

ÉCOLE DE TECHNOLOGIE SUPÉRIEURE  
UNIVERSITÉ DU QUÉBEC

MANUSCRIPT-BASED THESIS PRESENTED TO  
ÉCOLE DE TECHNOLOGIE SUPÉRIEURE

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR  
THE DEGREE OF DOCTOR OF PHILOSOPHY  
Ph.D.

BY  
George S. Eskander EKLADIOUS

A DISSIMILARITY REPRESENTATION APPROACH TO DESIGNING SYSTEMS FOR  
SIGNATURE VERIFICATION AND BIO-CRYPTOGRAPHY

MONTREAL, APRIL 29, 2014

© Copyright reserved

It is forbidden to reproduce, save or share the content of this document either in whole or in parts. The reader who wishes to print or save this document on any media must first get the permission of the author.

**BOARD OF EXAMINERS**

THIS THESIS HAS BEEN EVALUATED

BY THE FOLLOWING BOARD OF EXAMINERS:

Mr. Robert Sabourin, Thesis Director

Département de génie de la production automatisée à l'École de technologie supérieure

Mr. Eric Granger, Thesis co-Director

Département de génie de la production automatisée à l'École de technologie supérieure

Mr. Jean-Marc Robert, Committee President

Département de génie logiciel et des technologies de l'information à l'École de technologie supérieure

Mr. Luc Duong, Examiner

Département de génie logiciel et des technologies de l'information à l'École de technologie supérieure

Mr. Giuseppe Pirlo, External Examiner

Department of computer science at University of Bari

THIS THESIS WAS PRESENTED AND DEFENDED

IN THE PRESENCE OF A BOARD OF EXAMINERS AND PUBLIC

ON APRIL 17, 2014

AT ÉCOLE DE TECHNOLOGIE SUPÉRIEURE



## ACKNOWLEDGEMENTS

I am deeply thankful to my advisors, Dr. Robert Sabourin and Dr. Éric Granger. Their understanding and guidance, through more than four years of my doctoral study, are invaluable and indescribable by these simple words. I would like to thank them for their encouraging and friendly attitude and for every minute they spent in teaching me how to be a good researcher. Also thanks for the financial support they provided during this research.

I appreciate also the financial support provided by Natural Sciences and Engineering Research Council (NSERC) of Canada, BancTec Canada, and Ecole de Technologie Supérieure to complete this research.

I would like to express my sincere gratitude to the members of the examining committee: Dr. Jean-Marc Robert, Dr. Giuseppe Pirlo and Dr. Luc Duong, for evaluating this Thesis and providing valuable comments that improved this work.

I would like to thank all my friends and colleagues in the LIVIA (Laboratoire d'imagerie, de vision et d'intelligence artificielle), for sharing so many ideas. Thanks to: Fábio Dittrich, Idrissa Coulibaly, Jean-François Connolly, Jonathan Bouchard, Luana Batista, Mathias Adankon, M. Dewan, Marcelo Kapp, Miguel Gómora, Paulo Cavalin, Paulo Radtke, Philippe Lamontagne, Rafael Menelau, Saman Bashbaghi, and Wael Khreich.

Special thanks goes to Dominique Rivard for sharing his codes of the boosting feature selection algorithm, that was the base of some of the approaches reported in this Thesis. Thanks to Éric Thibodeau for sharing his codes for feature extraction that are used in this work. Special thanks to Christophe Pagano for the technical support he provided during this work. Also, special thanks to Eduardo Vellasques and César Alba for sharing their ideas during the start-up of this project. A special thanks goes to Dr. Alceu de Souza Britto who agreed to present some of my work at AFHA'13. I would like to express my deep gratitude to Bassem Guendy for his supportive and encouraging friendship, and helping in technical issues during my research work.

Special thanks to my wife Dina Girgis who shared me this long journey. Without her love, understanding, and support I would not even start this work. Her believe in me was the source of my success. To her I dedicate this Thesis. Thanks to my daughters Maria, Mira, and Gabriella for their love that have strengthen and inspired me.

Thanks to my parents Samir Eskander and Mary Zaki who supported me a lot during my whole life. Also, thanks to my mother-in-law Samia Farid for her love and support. I wish all of them would be here to receive my gratitude, but for their souls I send my gratitude. Thanks to my father-in-law Girgis Kamel for his love and support. Special thanks to my sister Lilian Samir and her husband Samy Bassit, and my brothers Joseph and John Eskander for their support, caring, and praying for me. Also special thanks to my sister-in-law Mira Girgis and her husband Ramez Hindy for their support and their sincere and valuable advices.

Last but not least, thanks to God for everything. May your name be exalted and glorified forever. The author of this work called George Eskander.

# **A DISSIMILARITY REPRESENTATION APPROACH TO DESIGNING SYSTEMS FOR SIGNATURE VERIFICATION AND BIO-CRYPTOGRAPHY**

George S. Eskander EKLADIOUS

## **ABSTRACT**

Automation of legal and financial processes requires enforcing of authenticity, confidentiality, and integrity of the involved transactions. This Thesis focuses on developing offline signature verification (OLSV) systems for enforcing authenticity of transactions. In addition, bio-cryptography systems are developed based on the offline handwritten signature images for enforcing confidentiality and integrity of transactions.

Design of OLSV systems is challenging, as signatures are behavioral biometric traits that have intrinsic intra-personal variations and inter-personal similarities. Standard OLSV systems are designed in the feature representation (FR) space, where high-dimensional feature representations are needed to capture the invariance of the signature images. With the numerous users, found in real world applications, e.g., banking systems, decision boundaries in the high-dimensional FR spaces become complex. Accordingly, large number of training samples are required to design of complex classifiers, which is not practical in typical applications. In contrast, design of bio-cryptography systems based on the offline signature images is more challenging. In these systems, signature images lock the cryptographic keys, and a user retrieves his key by applying a query signature sample. For practical bio-cryptographic schemes, the locking feature vector should be concise. In addition, such schemes employ simple error correction decoders, and therefore no complex classification rules can be employed.

In this Thesis, the challenging problems of designing OLSV and bio-cryptography systems are addressed by employing the dissimilarity representation (DR) approach. Instead of designing classifiers in the feature space, the DR approach provides a classification space that is defined by some proximity measure. This way, a multi-class classification problem, with few samples per class, is transformed to a more tractable two-class problem with large number of training samples. Since many feature extraction techniques have already been proposed for OLSV applications, a DR approach based on FR is employed. In this case, proximity between two signatures is measured by applying a dissimilarity measure on their feature vectors. The main hypothesis of this Thesis is as follows. The FRs and dissimilarity measures should be properly designed, so that signatures belong to same writer are close, while signatures of different writers are well separated in the resulting DR spaces. In that case, more cost-effective classifiers, and therefore simpler OLSV and bio-cryptography systems can be designed.

To this end, in Chapter 2, an approach for optimizing FR-based DR spaces is proposed such that concise representations are discriminant, and simple classification thresholds are sufficient. High-dimensional feature representations are translated to an intermediate DR space, where pairwise feature distances are the space constituents. Then, a two-step boosting feature selection (BFS) algorithm is applied. The first step uses samples from a development database,

and aims to produce a universal space of reduced dimensionality. The resulting universal space is further reduced and tuned for specific users through a second BFS step using user-specific training set. In the resulting space, feature variations are modeled and an adaptive dissimilarity measure is designed. This measure generates the final DR space, where discriminant prototypes are selected for enhanced representation. The OLSV and bio-cryptographic systems are formulated as simple threshold classifiers that operate in the designed DR space. Proof of concept simulations on the Brazilian signature database indicate the viability of the proposed approach. Concise DRs with few features and a single prototype are produced. Employing a simple threshold classifier, the DRs have shown state-of-the-art accuracy of about 7% AER, comparable to complex systems in the literature.

In Chapter 3, the OLSV problem is further studied. Although the aforementioned OLSV implementation has shown acceptable recognition accuracy, the resulting systems are not secure as signature templates must be stored for verification. For enhanced security, we modified the previous implementation as follows. The first BFS step is implemented as aforementioned, producing a writer-independent (WI) system. This enables starting system operation, even if users provide a single signature sample in the enrollment phase. However, the second BFS is modified to run in a FR space instead of a DR space, so that no signature templates are used for verification. To this end, the universal space is translated back to a FR space of reduced dimensionality, so that designing a writer-dependent (WD) system by the few user-specific samples is tractable in the reduced space. Simulation results on two real-world offline signature databases confirm the feasibility of the proposed approach. The initial universal (WI) verification mode showed comparable performance to that of state-of-the-art OLSV systems. The final secure WD verification mode showed enhanced accuracy with decreased computational complexity. Only a single compact classifier produced similar level of accuracy (AER of about 5.38 and 13.96% for the Brazilian and the GPDS signature databases, respectively) as complex WI and WD systems in the literature.

Finally, in Chapter 4, a key-binding bio-cryptographic scheme known as the fuzzy vault (FV) is implemented based on the offline signature images. The proposed DR-based two-step BFS technique is employed for selecting a compact and discriminant user-specific FR from a large number of feature extractions. This representation is used to generate the FV locking/unlocking points. Representation variability modeled in the DR space is considered for matching the unlocking and locking points during FV decoding. Proof of concept simulations on the Brazilian signature database have shown FV recognition accuracy of 3% AER and system entropy of about 45-bits. For enhanced security, an adaptive chaff generation method is proposed, where the modeled variability controls the chaff generation process. Similar recognition accuracy is reported, where more enhanced entropy of about 69-bits is achieved.

**Keywords:** Dissimilarity Representation, Dissimilarity Learning, Offline Signature Verification, Bio-Cryptography, Fuzzy Vault.



# **UNE APPROCHE BASÉE SUR LES VECTEURS DE REPRÉSENTATION PAR DIFFÉRENCES POUR LA CONCEPTION DES SYSTÈMES DE VÉRIFICATION DE SIGNATURES ET DE BIO-CRYPTOGRAPHIE**

George S. Eskander EKLADIOUS

## **RÉSUMÉ**

L'automatisation des processus légaux et financiers exige des transactions sécuritaires qui assurent l'authenticité, la confidentialité et l'intégrité de l'information. Cette thèse porte sur le développement de systèmes de vérification hors-ligne des signatures manuscrites (SVSM) dans le but de rehausser la sécurité des transactions. De plus, un système de bio-cryptographie est également proposé afin d'assurer la confidentialité et l'intégrité des transactions.

La conception des systèmes de vérification de signatures manuscrites représente un défi important étant donné la grande variabilité intra-classe et des similarités interclasses importantes. Les méthodes traditionnelles reposent sur une représentation de l'image de la signature dans des espaces de caractéristiques de grande dimension afin de capturer les invariants dans l'image de la signature. Dans un contexte d'applications bancaires, le nombre d'utilisateurs (classes) devient très important ce qui implique des frontières de décision complexes. De plus, une quantité importante de signatures sont requises pour l'apprentissage ce qui est impraticable pour les applications réelles. À l'opposé, la conception des systèmes de bio-cryptographie représente un défi encore plus grand. En effet, les images de signatures manuscrites servent pour bloquer les clés cryptographiques, conséquemment l'utilisateur utilise sa propre signature comme requête afin de récupérer sa clé. Les systèmes de bio-cryptographie imposent comme contrainte un très petit nombre de caractéristiques discriminantes. Enfin ces systèmes utilisent un décodeur qui utilise un mécanisme très simple de correction d'erreur, ce qui implique l'utilisation de règles de classification très simples.

Dans cette thèse, la conception des systèmes de vérification de signatures et de bio-cryptographie est basée sur l'utilisation des espaces de différences (DR, dissimilarity representation). Contrairement à l'utilisation des espaces de mesures, les approches basées sur les vecteurs de différences reposent sur la notion de mesure de proximité. Conséquemment, un problème de classification avec un grand nombre de classes devient un problème à deux classes dans l'espace des différences. Un avantage important est que le nombre de références devient très important avec une formulation dans l'espace des différences. Étant donné qu'un nombre important de caractéristiques est déjà disponible alors les espaces basés sur les différences sont faciles à construire. L'hypothèse formulée dans cette thèse est que les signatures d'un même scripteur sont semblables dans l'espace des caractéristiques ce qui se traduit par une projection proche de l'origine dans l'espace des différences. Conséquemment, les signatures de scripteurs différents sont projetées très loin de l'origine. Cette formulation du problème de vérification permet l'utilisation de classificateurs très simples basés sur des seuils de décision ce qui permet également la conception des systèmes de bio-cryptographie performants.

Le Chapitre 2 présente une approche pour l'optimisation des espaces DR qui permet une réduction importante de la dimensionnalité. Ces espaces optimisés sont également très discriminants ce qui permet l'utilisation de classificateurs très simples. Des espaces de caractéristiques de grande dimension sont transformés dans un espace intermédiaire composé d'un vecteur de différences mesurées deux à deux. Ensuite, une sélection de caractéristiques est effectuée en deux étapes par un algorithme de type Gentle Adaboost. La première étape consiste à définir un espace de représentation universel à partir d'une base de développement. Ensuite, une deuxième étape consiste à spécialiser l'espace de représentation pour chaque abonné au système de vérification. De plus, la représentation finale comprend une phase où la variabilité intra-classe est modélisée dans l'espace des différences et les prototypes les plus discriminants sont sélectionnés. Une contribution importante de l'approche proposée est que la conception des systèmes de bio-cryptographie et de vérification de signatures sont formulées de la même façon ce qui permet l'utilisation de simples classificateurs basés sur des seuils de décision dans l'espace des différences. Une preuve de concept est réalisée à partir de la base de signatures brésiliennes (PUCPR). Il résulte des espaces de représentation de très faible dimension (environ 20 caractéristiques) ce qui permet l'utilisation d'un seul prototype de référence par scripteur. Un taux d'erreur moyen de 7% est obtenu avec ce système, ce qui est compétitif avec les approches beaucoup plus complexes publiées dans la littérature.

Au chapitre 3, le problème de la conception des systèmes de vérification de signatures dans les espaces de différences est approfondi. La première version du système était performante, mais pas suffisamment sécuritaire. En effet, les prototypes de référence sont conservés en mémoire ce qui favorise une utilisation frauduleuse du système. Conséquemment, une nouvelle approche pour la conception de classificateurs dépendant du scripteur est proposée. Brièvement, la deuxième étape de traitement permet de revenir dans un espace de caractéristiques réduit ce qui favorise l'apprentissage d'un classificateur dans l'espace des mesures pour chaque scripteur. En conséquence, seul un classificateur est conservé en mémoire ce qui diminue les risques de fraudes pour les systèmes de vérification de signatures. Les résultats expérimentaux obtenus sur deux bases de signatures sont comparables à ceux publiés dans la littérature pour un classificateur beaucoup plus simple. Les performances moyennes obtenues sont de 5.38% et de 13.96% respectivement pour les bases PUCPR et GPDS.

Finalement, une première approche fonctionnelle et performante d'un système de bio-cryptographie basé sur les images de signatures manuscrites est présentée au Chapitre 4. Une méthode d'encapsulation de la clé cryptographique basée sur le Fuzzy Vault (FV) est détaillée dans ce chapitre. L'étape la plus importante et la plus innovante est la sélection de caractéristiques stables, discriminantes et adaptées pour le FV. La méthode de sélection de caractéristiques en deux étapes basée sur la représentation par différences et présentée au Chapitre 2 est reprise et adaptée pour la conception de systèmes de bio-cryptographie. De plus, une méthode originale pour la génération de points fantômes (chaff points) basée sur une mesure de variabilité dans l'espace des différences est également exposée. Une preuve de concept sur la base PUCPR montre un taux d'erreur moyen de 3% pour une valeur d'entropie de 45 bits. La méthode de génération de chaff points adaptative permet d'augmenter l'entropie du système à 69 bits avec la même performance.

**Mot-clés :** Représentations par différences, Apprentissage dans les espaces de différences, Vérification hors-ligne des signatures manuscrites, Bio-Cryptographie, Fuzzy Vault.



# CONTENTS

|  | Page |
|--|------|
| INTRODUCTION .....   | 1    |
| CHAPTER 1 BACKGROUND .....   | 11   |
| 1.1 Biometric systems .....  | 11   |
| 1.1.1 Performance of biometric systems .....                         | 11   |
| 1.1.2 Handwritten signature biometrics .....                         | 13   |
| 1.2 Bio-cryptography .....   | 14   |
| 1.2.1 Bio-cryptographic schemes .....                                | 15   |
| 1.2.2 Fuzzy Vault scheme .....                                       | 16   |
| 1.2.2.1 State of the art of FV .....                                 | 18   |
| CHAPTER 2 OPTIMIZED DISSIMILARITY REPRESENTATIONS .....              | 21   |
| 2.1 Introduction .....   | 22   |
| 2.2 From learning features to the dissimilarity representation ..... | 25   |
| 2.2.1 Learning feature representations .....                         | 26   |
| 2.2.2 Learning distance functions .....                              | 27   |
| 2.2.3 Learning dissimilarity representations .....                   | 28   |
| 2.2.4 Proposed approach .....  | 29   |
| 2.3 Dissimilarity Representation Optimization approach .....         | 30   |
| 2.3.1 Feature selection and dissimilarity learning .....             | 31   |
| 2.3.2 A two-step feature selection approach .....                    | 34   |
| 2.3.3 Adaptive dissimilarity measure .....                           | 36   |
| 2.3.4 Prototype selection in the dissimilarity space .....           | 37   |
| 2.4 Application to signature verification .....                      | 39   |
| 2.4.1 Dissimilarity-based signature verification .....               | 39   |
| 2.4.2 Applying the proposed approach .....                           | 40   |
| 2.5 Application to bio-cryptography .....                            | 41   |
| 2.5.1 Dissimilarity-based bio-cryptography .....                     | 41   |
| 2.5.2 Applying the proposed approach .....                           | 42   |
| 2.6 Experiments .....  | 44   |
| 2.6.1 Database .....   | 44   |
| 2.6.2 Class-independent optimization .....                           | 45   |
| 2.6.2.1 Feature extraction .....                                     | 45   |
| 2.6.2.2 Class-independent feature selection .....                    | 46   |
| 2.6.3 Class-specific optimization .....                              | 47   |
| 2.6.3.1 Class-specific feature selection .....                       | 47   |
| 2.6.3.2 Prototype selection in the D-space .....                     | 48   |
| 2.6.4 Performance evaluation .....                                   | 48   |
| 2.6.5 Results and discussion .....                                   | 51   |
| 2.7 Conclusion .....   | 54   |

|  |  |     |
|--|--|-----|
| 2.8  | Discussion .....   | 55  |
| CHAPTER 3 A HYBRID OFFLINE SIGNATURE VERIFICATION SYSTEM ..... |  | 57  |
| 3.1  | Introduction .....   | 57  |
| 3.2  | Pure WD and WI signature verification systems .....              | 59  |
| 3.3  | A Hybrid WI-WD signature verification system .....               | 62  |
| 3.3.1  | Theoretical basis .....  | 62  |
| 3.3.2  | System overview .....  | 66  |
| 3.3.3  | WI training .....  | 67  |
| 3.3.4  | WD training .....  | 69  |
| 3.3.5  | Signature verification .....                                     | 70  |
| 3.3.5.1  | WI-SV mode .....   | 70  |
| 3.3.5.2  | WD-SV mode .....   | 71  |
| 3.4  | Experimental methodology .....                                   | 71  |
| 3.4.1  | Signature databases .....  | 72  |
| 3.4.1.1  | Brazilian database .....   | 72  |
| 3.4.1.2  | GPDS database .....  | 72  |
| 3.4.2  | Feature extraction .....   | 73  |
| 3.4.3  | WI training .....  | 73  |
| 3.4.4  | WD training .....  | 74  |
| 3.4.5  | Performance measures .....                                       | 75  |
| 3.5  | Simulation results .....   | 76  |
| 3.5.1  | Performance of the WI and WD verification modes .....            | 77  |
| 3.5.1.1  | Brazilian database .....   | 77  |
| 3.5.1.2  | GPDS database .....  | 79  |
| 3.5.1.3  | Computational complexity .....                                   | 82  |
| 3.5.2  | Comparisons with systems in the literature .....                 | 84  |
| 3.5.2.1  | Brazilian database .....   | 84  |
| 3.5.2.2  | GPDS database .....  | 85  |
| 3.6  | Conclusions and future work .....                                | 87  |
| 3.7  | Discussion .....   | 88  |
| CHAPTER 4 A BIOMETRIC CRYPTOSYSTEM BASED ON SIGNATURES .....   |  | 89  |
| 4.1  | Introduction .....   | 89  |
| 4.2  | Fuzzy Vaults with offline signatures .....                       | 95  |
| 4.2.1  | Fuzzy Vault .....  | 95  |
| 4.2.1.1  | FV encoding .....  | 95  |
| 4.2.1.2  | FV decoding .....  | 97  |
| 4.2.2  | Encoding Fuzzy Vaults with signature images .....                | 99  |
| 4.3  | Selection of a user-specific signature representation .....      | 103 |
| 4.3.1  | Feature selection in the feature dissimilarity space .....       | 103 |
| 4.3.2  | A two-step BFS technique with dissimilarity representation ..... | 105 |
| 4.3.2.1  | Population-based feature selection .....                         | 107 |
| 4.3.2.2  | User-based feature selection .....                               | 109 |

|           |  |     |
|-----------|--|-----|
| 4.4       | A Fuzzy Vault system for offline signatures .....  | 110 |
| 4.4.1     | Enrollment process .....                           | 111 |
| 4.4.2     | Authentication process .....                       | 114 |
| 4.4.3     | Security analysis .....                            | 117 |
| 4.5       | Experimental results .....                         | 118 |
| 4.5.1     | Experimental methodology .....                     | 119 |
| 4.5.1.1   | Database .....                                     | 119 |
| 4.5.1.2   | Feature extraction .....                           | 119 |
| 4.5.1.3   | Feature selection .....                            | 120 |
| 4.5.1.4   | FV parameter values .....                          | 120 |
| 4.5.1.5   | Performance measures .....                         | 121 |
| 4.5.2     | Results on quality of feature representation ..... | 123 |
| 4.5.3     | Results on performance of the FV system .....      | 129 |
| 4.5.4     | Computational complexity .....                     | 133 |
| 4.6       | Conclusions and future work .....                  | 134 |
| 4.7       | Discussion .....                                   | 135 |
|           | GENERAL CONCLUSION .....                           | 137 |
| ANNEX I   | SCENARIOS FOR DR-BASED SV SYSTEMS .....            | 141 |
| ANNEX II  | ADAPTIVE CHAFF GENERATION .....                    | 149 |
| ANNEX III | TOWARDS AUTOMATED TRANSACTIONS .....               | 153 |
| ANNEX IV  | CASCADED SV-FV APPROACH .....                      | 167 |
|           | BIBLIOGRAPHY .....                                 | 172 |





## LIST OF TABLES

|   | Page |
|---|------|
| Table 2.1    The class-independent data set (CID) .....   | 46   |
| Table 2.2    The class-specific datasets (CSDs) .....   | 47   |
| Table 2.3    Average Hellinger distance over all users for the different design<br>scenarios .....        | 52   |
| Table 2.4    Impact of the prototype selection on average error rate over all users.....                  | 53   |
| Table 3.1    The Brazilian development database .....   | 73   |
| Table 3.2    The GPDS development database .....  | 73   |
| Table 3.3    The Brazilian WD database .....  | 74   |
| Table 3.4    The GPDS WD database .....   | 75   |
| Table 3.5    Overall error rates provided by systems designed by the Brazilian<br>database .....          | 83   |
| Table 3.6    Overall error rates provided by systems designed by the GPDS<br>database .....               | 86   |
| Table 4.1    FV parameter values .....  | 121  |
| Table 4.2    Signature samples of the first user in dataset $U$ .....                                     | 124  |
| Table 4.3    Average AUC over the 60 users in the dataset $U$ for the different<br>design scenarios ..... | 126  |
| Table 4.4    FV performance for different key sizes .....   | 128  |
| Table 4.5    Impact of chaff quantity on the FV performance .....   | 129  |
| Table 4.6    Impact of using a user password as a second authentication measure .....                     | 131  |
| Table 4.7    Performance of the baseline SV systems and the proposed FV<br>system.....                    | 132  |
| Table 4.8    Comparison of the baseline SV Systems and the proposed FV<br>system.....                     | 133  |



## LIST OF FIGURES

|            |  | Page |
|------------|--|------|
| Figure 1.1 | Illustration of the FV locking/unlocking process. ....   | 17   |
| Figure 2.1 | Illustration of a dissimilarity representation (DR) built on top of a feature representation (FR) .....                | 31   |
| Figure 2.2 | Illustration of feature selection in the original feature space $F$ and in the feature-dissimilarity space $FD$ . .... | 32   |
| Figure 2.3 | A framework of the optimizing dissimilarity representations approach .....   | 35   |
| Figure 2.4 | Illustration of the transformation from the feature-dissimilarity space to the dissimilarity space .....               | 37   |
| Figure 2.5 | Proposed model of the FV functionality .....   | 43   |
| Figure 2.6 | Dissimilarity score distribution for a specific user. ....   | 51   |
| Figure 2.7 | Dissimilarity score distribution for different forgery types. ....   | 51   |
| Figure 2.8 | Dissimilarity score distributions for different prototypes. ....   | 53   |
| Figure 3.1 | Illustration of feature selection in the original feature space and in the feature-dissimilarity space. ....           | 62   |
| Figure 3.2 | Hybrid WI-WD SV system .....   | 66   |
| Figure 3.3 | Average AUC of ROC curves for the WI and WD classifiers for the Brazilian database .....                               | 77   |
| Figure 3.4 | AER for the WI and WD classifiers for the Brazilian database .....   | 78   |
| Figure 3.5 | FRR and FAR for the WD-SV mode for the Brazilian database .....  | 78   |
| Figure 3.6 | Average AUC of ROC curves for the WI and WD classifiers for the GPDS database .....                                    | 79   |
| Figure 3.7 | ROC curves for the WI and WD modes for different training sizes (user 1) .....   | 80   |
| Figure 3.8 | ROC curves for the WI and WD modes for different training sizes (user 2) .....   | 80   |

|             |   |     |
|-------------|---|-----|
| Figure 3.9  | AER for WD-SV verification mode, for the GPDS database .....  | 81  |
| Figure 3.10 | FRR and FAR for the WD-SV mode for the GPDS database.....   | 82  |
| Figure 4.1  | Block diagram of the FV encoding process .....  | 95  |
| Figure 4.2  | Block diagram of the FV decoding process .....  | 96  |
| Figure 4.3  | Illustration of the chaff filtering Process .....   | 98  |
| Figure 4.4  | Illustration of FV encoding with signature images.....  | 100 |
| Figure 4.5  | Chaff filtering in the feature encoding space .....   | 101 |
| Figure 4.6  | Chaff filtering in the polynomial space .....   | 101 |
| Figure 4.7  | Illustration of feature selection in the feature dissimilarity space<br>and in the feature encoding space ..... | 104 |
| Figure 4.8  | Overall block diagram of the proposed approach including<br>population and user-based feature selection .....   | 107 |
| Figure 4.9  | Block diagram of the proposed FV system for offline signature<br>images. ....                                   | 110 |
| Figure 4.10 | Block diagram of the proposed FV encoding process.....  | 112 |
| Figure 4.11 | Illustration of the chaff generation process .....  | 113 |
| Figure 4.12 | Proposed FV Decoding .....  | 115 |
| Figure 4.13 | Proposed chaff filtering with AMW .....   | 116 |
| Figure 4.14 | Illustration of chaff filtering in feature encoding space for the first<br>user in the dataset $U$ .....        | 124 |
| Figure 4.15 | Similarity score distribution for the first user in dataset $U$ .....   | 126 |
| Figure 4.16 | ROC curve for the first user in the dataset $U$ .....   | 127 |
| Figure 4.17 | AUC for the 60 Users in the dataset $U$ .....   | 127 |
| Figure 4.18 | Trade-off between FV security and recognition measures. ....  | 128 |
| Figure 4.19 | Trade-off between FV security and average recognition error rates. ....   | 129 |

## LIST OF ABBREVIATIONS

|       |   |
|-------|---|
| OLSV  | Offline Signature Verification            |
| FR    | Feature Representation                    |
| DR    | Dissimilarity Representation              |
| SV    | Signature Verification                    |
| WI    | Writer-Independent                        |
| WD    | Writer-Dependent                          |
| FV    | Fuzzy Vault                               |
| CIR   | Class-Independent Representation          |
| CSR   | Class-Specific Representations            |
| BFS   | Boosting Feature Selection                |
| WC    | Within-Class                              |
| BC    | Between-Class                             |
| KNN   | K-Nearest Neighbors                       |
| SVM   | Support Vector Machines                   |
| WI-SV | Writer-Independent Signature Verification |
| WD-SV | Writer-Dependent Signature Verification   |
| DB    | Database                                  |
| CID   | Class-Independent Dataset                 |
| CSD   | Class-Specific Data Set                   |
| FD    | Feature Dissimilarity                     |

|      |  |
|------|--|
| HMMs | Hidden Markov Models                     |
| ESC  | Extended Shadow Code                     |
| FAR  | False Accept Rate                        |
| GAR  | Genuine Accept Rate                      |
| ROC  | Receiver Operating Characteristic        |
| AUC  | Area Under the ROC Curve                 |
| DPDF | Directional Probability Density Function |
| DS   | Decision Stump                           |
| P    | Population-Based Representation          |
| U    | User-Based Representation                |
| AER  | Average Error Rate                       |
| TFV  | Total Number of Feature Values           |
| EER  | Equal Error Rate                         |
| IPV  | Intra-Personal Variability               |
| IPS  | Inter-Personal Similarity                |
| PR   | Population-Based Representation          |
| UR   | User-Based Representation                |
| AMW  | Adaptive Matching Window                 |
| RS   | Reed-Solomon Code                        |
| P-FS | Population-Based Feature Selection       |
| U-FS | User-Based Feature Selection             |

|     |                             |
|-----|-----------------------------|
| BCT | Bio-Cryptography Template   |
| GF  | Galois Field                |
| FRR | False Reject Rate           |
| PKI | Public Key Infrastructure   |
| ECC | Elliptic Curve Cryptography |





## LIST OF SYMBOLS

|                             |   |
|-----------------------------|---|
| $x_i$                       | Feature representation of a questioned sample   |
| $x$                         | Feature representation of a prototype sample  |
| $D_{\mathbf{Q}}$            | Metric distance function defined by a postive definite matrix (kernel) $Q$                                |
| $\mathbf{Q}^T$              | Metric tensor for a class $T$   |
| $U$                         | Number of classes   |
| $u$                         | A template class index  |
| $R$                         | Number of prototypes per class  |
| $r$                         | A prototype index   |
| $p_{ur}$                    | A prototype number $r$ for a class $u$  |
| $v$                         | A questioned class index  |
| $J$                         | Number of questioned samples per class  |
| $Q_{vj}$                    | A questioned sample number $j$ from a class $v$   |
| $\delta^{Q_{vj}p_{ur}}$     | Dissimilarity between a questioned sample $Q_{vj}$ and a prototype $p_{ur}$                               |
| $N$                         | Dimensionality of an optimized class-specific feature and feature dissimilarity spaces                    |
| $n$                         | Index of a specific dimension of an optimized feature and feature dissimilarity spaces                    |
| $f_n^{Q_{vj}}$              | Feature number $n$ for a questioned sample $Q_{vj}$   |
| $f_n^{p_{ui}}$              | Feature number $n$ for a prototype sample $p_{ui}$  |
| $\mathbf{d}^{Q_{vj}p_{ur}}$ | Dissimilarity vector represent the dissimilarity of a questioned sample $Q_{vj}$ and a prototype $p_{ur}$ |

|                             |  |
|-----------------------------|--|
| $\delta f_n^{Q_{vj}p_{ur}}$ | Dissimilarity between a questioned sample $Q_{vj}$ and a prototype $p_{ur}$ measured by means of feature $f_n$ , or erasures error due to false mismatch in bio-cryptography |
| $g$                         | Distance function  |
| $\delta_E^{Q_{vj}p_{ur}}$   | Euclidean distance between a questioned sample $Q_{vj}$ and a prototype $p_{ur}$   |
| $\delta f_n$                | Dissimilarity feature $n$  |
| $M$                         | Dimensionality of the original (not optimized) feature and feature dissimilarity spaces  |
| $L$                         | Dimensionality of the class-independent optimized feature and feature dissimilarity spaces   |
| $R1$                        | Original number of prototypes per class  |
| $R2$                        | Optimized number of prototypes per class   |
| $i$                         | Boosting iteration   |
| $\delta f_i$                | Dissimilarity feature selected at a boosting iteration $i$   |
| $DS_i$                      | Decision stump trained at a boosting iteration $i$   |
| $\Delta$                    | Feature dissimilarity vector   |
| $\delta_A^{Q_{vj}p_{ur}}$   | Adaptive dissimilarity measure between a questioned sample $Q_{vj}$ and a prototype $p_{ur}$   |
| <b>P</b>                    | Vector of prototypes   |
| <b>D</b> $^{Q_{vj}}$        | Dissimilarity vectors of a questioned sample $Q_{vj}$  |
| $\epsilon$                  | Dissimilarity threshold  |
| $SV$                        | Signature verification function  |

|                           |  |
|---------------------------|--|
| $K$                       | Cryptographic key  |
| $F^{Q_{vj}}$              | Unlocking message by a query signal $Q_{vj}$   |
| $F^{p_{ur}}$              | Locking message by a prototype signal $p_{ur}$   |
| $\delta'$                 | Noise error due to false matching with chaffs  |
| $r^*$                     | Index of the best prototype  |
| $FV$                      | Fuzzy Vault function   |
| $U^{\text{CID}}$          | Number of classes in the class independent dataset   |
| $R_1^{\text{T}}$          | Number of prototypes per class in the training set   |
| $R_1^{\text{V}}$          | Number of prototypes per class in the validation set   |
| $U^{\text{CSD}}$          | Number of classes in the class specific dataset  |
| $R_1^{\text{cs}}$         | Number of prototypes per class for the class-specific dataset  |
| $\delta_S^{Q_{vj}p_{ur}}$ | Strict dissimilarity measure between a questioned sample $Q_{vj}$ and a prototype $p_{ur}$                   |
| $H(WC, BC)$               | Hellinger distance between within-class and between-class dissimilarity clusters                             |
| $\mu_1$                   | Mean value of the within-class dissimilarities   |
| $\mu_2$                   | Mean value of the between-class dissimilarities  |
| $\sigma_1$                | Variance of the within-class dissimilarities   |
| $\sigma_2$                | Variance of the between-class dissimilarities  |
| $H_{\text{random}}$       | Hellinger distance between within-class (for random forgeries only) and between-class dissimilarity clusters |

|                   |  |
|-------------------|--|
| $H_{simple}$      | Hellinger distance between within-class (for simple forgeries only) and between-class dissimilarity clusters                   |
| $H_{simulated}$   | Hellinger distance between within-class (for simulated forgeries only) and between-class dissimilarity clusters                |
| $H_{all}$         | Hellinger distance between within-class (for all forgery types) and between-class dissimilarity clusters                       |
| $\hat{H}$         | Hellinger distance averaged over all users   |
| $AER$             | Average error rate   |
| $FRR$             | False reject rate  |
| $FAR_{random}$    | False accept rate for random forgeries   |
| $FAR_{simple}$    | False accept rate for simple forgeries   |
| $FAR_{simulated}$ | False accept rate for simulated forgeries  |
| $M_G$             | Multi-feature representations extracted from genuine signature   |
| $M_F$             | Multi-feature representations extracted from forgery signature   |
| $S^Q$             | Genuine signature  |
| $S^F$             | Forgery signature  |
| $M_i$             | Multi-feature representation of a signature $i$  |
| $D_{ij}^M$        | Dissimilarity representation of signatures $i$ and $j$ , built over multi-feature representation $M$                           |
| $d_n$             | Best dimension at boosting iteration $n$   |
| $DS_n$            | Decision stump trained at boosting iteration $n$   |
| $p_n^{left}$      | Confidence of a decision taken by a decision stump $DS_n$ , when the feature value lies to the left of the splitting threshold |

|               |   |
|---------------|---|
| $p_n^{right}$ | Confidence of a decision taken by a decision stump $DS_n$ , when the feature value lies to the right of the splitting threshold |
| $H^{wi}$      | Decision boundary of a writer-independent classifier  |
| $DS_n^{wi}$   | Decision stump of a writer-independent classifier, designed at a boosting iteration $n$   |
| $S^E$         | Enrolling signatures set  |
| $S^F$         | Forgery signatures set used for training a writer-dependent classifier  |
| $P_G$         | Population representation extracted from genuine signature  |
| $P_F$         | Population representation extracted from forgery signature  |
| $H^{wd}$      | Decision boundary of a writer-dependent classifier  |
| $B^{wi}$      | The Brazilian development database  |
| $g^{wi}$      | The GPDS development database   |
| $T^{wi}$      | Number of decision stumps in the writer-independent system  |
| $B^{wd}$      | The Brazilian testing database  |
| $g^{wd}$      | The GPDS testing database   |
| $T^{wd}$      | Number of decision stumps in the writer-dependent system  |
| $P$           | Population space  |
| $AUC$         | Area under ROC curve  |
| $S^Q$         | Questioned signature  |
| $TFV$         | Total number of feature values  |
| $nc$          | Number of partial classification decisions of a classifier  |

XXX

|  |  |
|--|--|
| $N_i$                                  | Number of features per sample processed by a classifier $i$  |
| $S_i$                                  | Number of signature samples processed by a classifier $i$  |
| $WI - SV$                              | Writer-independent signature verification function   |
| $WD - SV$                              | Writer-dependent signature verification function   |
| $D_{QE}^P$                             | Dissimilarity representation of the query sample $S^Q$ built on top of the population-based feature representation $P$ |
| $U$                                    | User-specific space  |
| $U_Q$                                  | User-specific representation of a query signature $Q$  |
| $K$                                    | Cryptographic key  |
| $T$                                    | Biometric template   |
| $V_T$                                  | Fuzzy Vault encodes a template $T$   |
| $F_T$                                  | Feature representation extracted from a template $T$   |
| $t$                                    | Number of elements encoded in a FV   |
| $\mathbf{A}$                           | Locking set  |
| $a_i$                                  | Locking point quantized value of the feature $f_i$   |
| $p$                                    | Locking polynomial   |
| $k$                                    | Degree of the polynomial degree  |
| $\mathbf{C}$                           | Polynomial coefficient vector  |
| $GF(2^l)$                              | Finite Galois field of $l$ bits  |
| $(\mathbf{A}, p(\mathbf{A}))$          | FV genuine points  |
| $(\hat{\mathbf{A}}, \hat{\mathbf{P}})$ | FV chaff points  |

|  |  |
|--|--|
| $\hat{a}$                                  | Chaff point  |
| $(\tilde{\mathbf{A}}, \tilde{\mathbf{P}})$ | FV all points  |
| $F^Q$                                      | Feature representation extracted from a query $Q$                                  |
| $\mathbf{B}$                               | Unlocking set  |
| $b_i$                                      | Unlocking point- quantized value of the feature $f_i^Q$ extracted from a query $Q$ |
| $\bar{\mathbf{A}}$                         | Matching set   |
| $t'$                                       | Length of the matching set   |
| $\bar{N}$                                  | Length of subsets of length $k + 1$ in the matching set                            |
| $SS^Q$                                     | Similarity score of a query $Q$  |
| $D\Theta$                                  | Decoding Threshold   |
| $mw_i$                                     | Matching window for a FV element $i$   |
| $\mathbf{FI}$                              | Indexes of selected features   |
| $FI_i$                                     | Index of a feature $i$   |
| $\mathbf{VI}$                              | Virtual indexes of selected features   |
| $VI_i$                                     | Virtual index of a feature $i$   |
| $\mathbf{T}$                               | Set of signature templates   |
| $\mathbf{UR}$                              | User representations matrix  |
| $PW$                                       | Password   |
| $BCT$                                      | Bio-Cryptography Template  |
| $\mathbf{X}^T$                             | FV locking set extracted from a template $T$                                       |
| $\mathbf{Y}^Q$                             | FV unlocking set extracted from a template $Q$                                     |

|                |  |
|----------------|--|
| $l$            | Quantization size                        |
| $\mathbf{G}_1$ | Group 1 chaff set                        |
| $\mathbf{G}_2$ | Group 2 chaff set                        |
| $\Omega$       | Chaff separation                         |
| $\alpha$       | Chaff groups ratio                       |
| $g_1$          | Amount of chaff features belong to $G_1$ |
| $g_2$          | Amount of chaff features belong to $G_2$ |
| $r$            | FV size                                  |
| $z$            | Total number of chaffs                   |
| $p'$           | Reconstructed polynomial                 |
| $K'$           | Reconstructed cryptographic key          |
| $KS$           | Length of cryptographic key              |
| $\mathbf{R}$   | Reference subset                         |
| $\mathbf{Q}$   | Questioned subset                        |
| $\mathbf{P}$   | Population-based dataset                 |
| $\mathbf{U}$   | User-based dataset                       |
| $e$            | Public key                               |
| $q$            | Public parameter                         |
| $E$            | Signature message                        |
| $DigSig$       | Digital signature                        |



## INTRODUCTION

Automation of legal and financial processes requires enforcing of authenticity, confidentiality, and integrity of the involved transactions. For the paper-based processes, handwritten signature is the most universally accepted method of authentication, and offline signature verification (OLSV) systems are developed to automate this security issue (Impedovo and Pirlo, 2008). However, the other security issues are not fulfilled within manual systems. For instance, confidentiality of a document is lost when it is accessed by an intruder. In addition, integrity of a document is not guaranteed, as its content might be altered by impostors after being issued and signed by its producer.

Confidentiality and integrity of electronic documents maybe achieved through the encryption and digital signature cryptographic schemes, respectively. These schemes rely on long cryptographic keys that are usually stored on, e.g., smart cards, and they are accessed by shorter, easy to remember, passwords. Using biometrics (physical or behavioral human traits) like fingerprint may replace the traditional passwords for more trusted user authentication, by implementing the so-called bio-cryptographic schemes (Uludag *et al.*, 2004).

Despite the intensity of research on bio-cryptography based on more physiological traits like fingerprints, iris, face, etc., there is no conclusive research on more behavioral traits such as offline handwritten signature images.

Developing a reliable bio-cryptography system based on the offline handwritten signatures might enforce confidentiality and integrity of the automated transactions. In this case, automating the existing paper-based processes is transparent to users, as they continually employ their handwritten signatures and they become isolated from the details of the new technology, and its related security issues.

This Thesis focuses on developing accurate, simple, and secure OLSV and signature-based bio-cryptography systems, so automation of legal and business processes becomes possible.

Standard OLSV and bio-cryptography systems are designed in the feature representation (FR) space, where discriminative features are extracted into patterns that can be viewed as points in the feature space. With the numerous classes (writers), high dimensional representations, and with a limited number of training samples per class, design of an efficient FR becomes impractical (Guyon and Elisseeff, 2003). Indeed it is hard to extract many reference signatures from a person for enrollment. In addition, designing FRs and FR-based classifiers using signature samples of existing users, does not necessarily produce systems that generalize for unseen users and samples during operations. In practice, it is impossible to locate a FR space in which signatures of all current and future users share the same distribution. The dissimilarity concept, where samples that belong to same class should be similar, while samples that come from different classes should be dissimilar, provides a solution. Instead of learning representations of specific unseen signatures, which is impossible, it is possible to learn a generic model of the intra-personal and inter-personal variabilities using a huge number of signature samples extracted from a development database.

In this Thesis, the design of OLSV and signature-based bio-cryptography systems, is tackled by employing the dissimilarity representation (DR) approach (Pekalska and Duin, 2002). Instead of designing classifiers in the feature space, the DR approach provides a classification space that is defined by some proximity measure.

It is argued that the core task of biometric authentication is actually a multiple classification problem in the sense that it solves several authentication tasks simultaneously (Bengio and Marithoz, 2007). As a result, there is a need of a joint learning among the individual authentication systems. Some approaches designed multiple single-user classifiers jointly, like parameter sharing among several classifiers (Reynolds, 2000). In the context of offline signature verification, the DR approach provides a way to learn dissimilarity ranges of the intra-personal and the inter-personal signature samples. During verification, two signature samples are matched by comparing the difference in their representations to the modeled dissimilarity ranges. Moreover, computing proximities between signature images enlarges the number of

samples available for training, and avoids the curse-of-dimensionality problem (Rivard *et al.*, 2013).

Since many feature extraction techniques are already proposed for the OLSV application, we employed a FR-based DR approach where the DR space is build on top of a FR space (Duin *et al.*, 2010). In this case, proximity between two signatures (a query and a prototype) is measured by applying a certain dissimilarity measure on their feature vectors. The FRs and dissimilarity measures should be properly designed, so that signatures belong to same writer are close, while signatures of different writers are well separated in the resulting DR spaces. In that case, simple classification rules, e.g., thresholds, might be sufficient, and simple OLSV and bio-cryptography systems can be designed based on these representations.

### **Problem statement**

Design of the OLSV systems is challenging, as signatures are more behavioral biometrics that have intrinsic intra-personal variations and inter-personal similarities. Static features extracted from the offline signature images may incorporate a lower level of stability and discrimination than that with the online signatures, where dynamic signals, e.g., pressure, velocity, etc., are acquired during the signing process. Standard signature verification (SV) systems are designed in the Feature Representation (FR) space. For OLSV systems, high-dimensional feature representations are needed to capture the variations of the signature images. With the numerous users, as found in real world applications, e.g., banking systems, decision boundaries in the high-dimensional FR spaces become complex. Enough training samples are needed to learn such complex models and collecting these samples is not practical with typical SV applications.

Two main techniques are proposed for OLSV, namely, writer-dependent (WD) and writer-independent (WI) systems. These techniques have shown a compromise between security, accuracy, and complexity of the produced OLSV systems. For the WD-SV systems, a single classifier is designed for each user using his reference signatures. Such systems are secure, as no templates are stored for verification. However, it is not easy to achieve high recognition accuracy, when few number of reference samples are available for training. In addition, models

that are designed based on samples of users enrolled during the design phase, might be invalid when other users are added to the system. Some authors proposed complex systems, e.g., multi-classifier systems, dynamic selection of classifiers, etc., for enhanced recognition accuracy (Batista *et al.*, 2012).

On the other hand, the WI-SV technique produces a single classifier for the whole population. Signature samples from a development database are used for training, while the classifier is exploited on real users whose signatures are not seen during the design phase. Although WI systems provide better generalization for unseen users, they are still complex, and insecure as templates are stored for verification (Rivard *et al.*, 2013).

Design of bio-cryptography systems based on the offline signature images is more challenging than designing classical OLSV systems. In these systems, signature images lock the cryptographic keys, and a user retrieves his key by applying a query signature sample. For practical bio-cryptographic schemes, the locking feature vector should be concise, and the limited representation might not capture the variance of the intra-personal and inter-personal classes. In addition, such schemes employ simple error correction decoders, and therefore no complex classification rules can be employed. These systems also involve similar trade-offs between security, accuracy, and complexity, like that with classical OLSV systems.

Reliable bio-cryptographic schemes run in a key-binding mode, where both cryptographic and biometric keys are coupled in a way that neither one can be decoupled without using a genuine biometric query sample. The most commonly used key-binding scheme is the Fuzzy Vault (FV). It relies on embedding chaff (noise) information to hide the genuine information extracted from the biometric signal (Juels and Sudan, 2002). Security of a FV depends on the amount of embedded chaffs. However, in case that chaffs interfere with the genuine biometric information, accuracy of the FV decoder degrades. Due to high variability of the offline signature images, it is not easy to extract stable genuine information that do not interfere with the chaffs, and this requires a trade-off between security and accuracy of the designed FV.

Tackling these design challenges using a DR approach aims to use a limited number of signature templates (available for training) to produce a concise, stable, and discriminant user-specific DR space, in which accurate, simple, and secure OLSV and FV bio-cryptography systems are developed. This approach leads to the following main research questions:

- a. Using a limited user-specific training set, how to generate a concise FR space that is stable for the specific user, and its discriminating power generalizes for unseen users (compromising advantageous of WD and WI techniques)?
- b. How to design a proximity measure that alleviates the intrinsic variability of the offline signature images?
- c. How to select efficient prototypes that produce the DR space?
- d. How to formulate the OLSV and the FV systems as simple classifiers in the DR space?
- e. How to generate enough chaffs without significantly degrading the FV accuracy?

### **Objective and contributions**

The main objective of this Thesis is to develop reliable SV and bio-cryptographic systems that enable automating legal and business processes. As most of the existing processes are paper-based and they employ handwritten signature images for authentication, so we design our systems for the offline handwritten signatures. Physical presence of persons is not mandatory in case of the OLSV systems, so it can be applied in a broader range of applications than the online SV systems. Since both FR and DR spaces are exploited in the literature for designing WD and WI systems, respectively, and they have shown trade-offs between performance measures, we propose a hybrid technique as a compromise of the two approaches. Different than the DR approaches found in the literature, that focus on designing classifiers in a defined DR, we focus on learning the DR itself.

The main contribution of this Thesis is a DR optimization approach that learns a reliable DR build on top of a concise and discriminant FR. Resulting DRs are designed so that a global

class-independent representation (CIR) represents all, and even classes that are unseen during the design phase, where it could be tuned for specific classes by means of class-specific training data. Since the proposed approach involves feature selection, it can also be considered as a tool to design pure feature-based classifiers. Moreover, as it involves distance function learning and prototype selection, the approach can be employed to design distance-based classifiers that work in the FR space.

First, a reliable DR is designed through employing the proposed DR optimization approach, and the OLSV and bio-cryptography systems are formulated as simple classifiers in the resulting space. Then, the OLSV system is enhanced, by considering the proposed optimization approach as a tool for feature selection. The designed DRs are translated to a reduced and discriminant FR space, where more secure, simpler, and more accurate OLSV classifiers are designed. Finally, a complete bio-cryptographic FV implementation is developed based on the designed DR.

In the first contribution, high-dimensional FRs are translated to an intermediate DR space, where pairwise feature distances are the space constituents. Then, a two-step boosting feature selection (BFS) algorithm is applied (Tieu and Viola, 2004). The first step uses samples from a development database, and aims to producing a universal space of reduced dimensionality. The resulting universal space is further reduced and tuned for specific users through a second BFS step using user-specific training set. In the resulting space, feature variations are modeled and an adaptive dissimilarity measure is designed. This measure generates the final DR space, where discriminant prototypes are selected for enhanced representation. It was demonstrated that concise representations produced separable clusters in the dissimilarity space. Accordingly, a simple threshold provides a high level of accuracy comparable to complex OLSV systems in the literature. Moreover, the bio-cryptography design problem is formulated as a traditional classifier in the produced DR space, where designing such systems is more tractable.

In the second contribution, the OLSV problem is further studied. Although the aforementioned SV implementation has shown acceptable recognition accuracy, the resulting systems are not secure as signature templates must be stored for verification. For enhanced security,

the previous implementation is modified as follows. The first BFS step is implemented as aforementioned, producing a writer-independent (WI) system. This enables starting system operation, even if users provide a single signature sample in the enrollment phase. However, the second BFS is modified to run in a FR space instead of a DR space, so that no signature templates are used for verification. To this end, the universal space is translated back to a FR space of reduced dimensionality, so that designing a writer-dependent (WD) system by the few user-specific samples is tractable in the reduced space. It was demonstrated that the secure WD verification mode showed enhanced accuracy with decreased computational complexity than that of the universal WI verification mode.

In the third contribution, a FV bio-cryptographic scheme is implemented based on the offline signature images. The proposed DR-based two-step BFS technique is employed for selecting a compact and discriminant user-specific FR from a large number of feature extractions. This representation is used to generate the FV locking/unlocking points. In the encoding phase, the locking points lock user cryptographic key in a FV. During decoding, the unlocking points are used to unlock user key from his FV. Representation variability modeled in the DR space is considered for matching the unlocking and locking points during FV decoding. Proof of concept simulations have shown an acceptable compromise of recognition accuracy and system entropy of the designed FVs. For enhanced security, an adaptive chaff generation method is proposed, where the modeled variability controls the chaff generation process. Similar recognition accuracy is reported with a higher security level.

### **Organization of this Thesis**

This manuscript-based Thesis is organized into four chapters. In Chapter I the areas of biometrics, handwritten signature biometrics, bio-cryptography, and the fuzzy vault scheme are presented and reviewed.

In Chapter II the DR optimizing approach is presented. This approach is generic as it might be applied for different pattern classification problems that are designed with limited data from many classes, then they adapt whenever new data becomes available, e.g., adaptive biometric

systems. Validating the approach on the OLSV and bio-cryptography systems design problems has demonstrated its viability. Some concepts of applying this approach for designing OLSV verification systems was published at International workshop on Automated Forensic Handwriting Analysis (Eskander *et al.*, 2013a). Also, applicability of this approach to the bio-cryptographic system design was published the 2<sup>nd</sup> International workshop on Similarity-Based Pattern Analysis and Recognition (Eskander *et al.*, 2013f). The complete approach with applications on OLSV and bio-cryptography was submitted to the special issue of the IEEE Transactions on Neural Networks and Learning Systems on "Learning in non-(geo)metric spaces" (Eskander *et al.*, 2013e).

In Chapter III the DR optimization approach is employed to provide a solution for compromising between pure WD and WI techniques for OLSV. A universal WI classifier is designed with a development database, to enable starting system operation with few signature templates. Switching to a more secure, less complex, and more accurate WD operational mode is possible whenever enough samples are collected for a specific user. Adaptation of the WI classifiers to specific users is achieved through tuning the universal signature representation to each user, while training his WD classifier. Simulation results on two real-world offline signature databases demonstrated the feasibility and robustness of the proposed solution. The initial universal (WI) verification mode showed comparable performance to that of state of the art OLSV systems. The final user-specific WD verification mode showed enhanced accuracy with decreased computational complexity. Only a single compact classifier produced similar level of accuracy as complex WI and WD systems in literature. In addition, the produced WD classifiers are more secure than the baseline WI classifiers, eliminating the need to store user templates for verification. Preliminary version of this solution was published at the 13th International conference on Frontiers in Handwriting Recognition (Eskander *et al.*, 2012). The complete system design and analysis was published at the IET-Biometrics Journal, Special issue on Handwriting Biometrics (Eskander *et al.*, 2013c).

In Chapter IV the DR optimization approach is applied to design a complete FV bio-cryptography system based on offline signature images. A few number of features is selected based on the



designed DR space, and used to produce the bio-cryptographic tokens. The feature variability that is learned in the DR space are used for decoding the tokens and release the keys for genuine users. It is shown that selecting features based on signatures from an independent (development) database could represent the actual system users. While, running another user-specific feature selection process enhanced the quality of feature representation. Also, adapting the features matching window based on their expected variations results in better FV performance. A user password is used as a second authentication measure to enhance FV system accuracy. For further enhancement of the FV recognition performance, a simple ensemble of FVs is produced through applying the majority vote decision fusion concept. A preliminary version of this system was published at the IEEE Workshop on Computational Intelligence and Identity Management (Eskander *et al.*, 2011). The complete system implementation and performance was published at Information Sciences (Eskander *et al.*, 2014a).

In Appendix I we explore different scenarios for employing the DR approach for replacing and/or enhancing the standard SV systems. A general framework for designing FR/DR based systems is proposed where the DR approach can be applied in different scenarios. This framework might enable the design of a new family of classification systems, such as global and hybrid global/user-specific classifiers. Also, the proposed framework suggests employing the DR approach as an intermediate design tool for enhanced performance of standard feature-based systems. Content of this appendix was published at the International workshop on Automated Forensic Handwriting Analysis (Eskander *et al.*, 2013a).

In Appendix II an adaptive FV chaff generation method is proposed, where the modeled variability controls the chaff generation process. Similar recognition accuracy is reported, where more enhanced entropy is achieved. This method was published at the International workshop on Emerging Aspects in Handwritten Signature Processing (Eskander *et al.*, 2013b)

In Appendix III the designed signature-based FV implementation is employed to produce digital signatures using off-line handwritten signatures. This methodology facilitates the automation of business processes, where users continually employ their handwritten signatures for authentication. First, signature templates from a user are captured and employed to lock his

private key in a FV. Then, when the user signs a document by hand, his handwritten signature image is employed to unlock his private key. The unlocked key produces a digital signature that is attached to the digitized document. Verification of the digital signature by a recipient implies authenticity of the manuscript signature and integrity of the signed document. Experimental results confirms the viability of the proposed approach. The content of this appendix was published at the 9th International conference on Machine Learning and Data Mining (Eskander *et al.*, 2013d).

In Appendix IV a novel approach is proposed for enhancing the accuracy of signature-based biometric cryptosystems. Instead of using an additional password for enhanced security, the same signature sample is processed by a SV classifier before triggers the FV decoders. Using this cascaded approach, the high FAR of FV decoders is alleviated by the higher capacity of SV classifiers to detect impostors. The content of this appendix is published at the 14th International Conference on Frontiers in Handwriting Recognition (ICFHR-2014) (Eskander *et al.*, 2014b).

## **CHAPTER 1**

### **BACKGROUND**

#### **1.1 Biometric systems**

Biometrics is the science of recognizing an individual based on his physiological or behavioral traits (Anil K. Jain, 2004). Examples of more physiological traits are: face, fingerprint, hand geometry, iris, retina, ear, and DNA. Examples of more behavioral traits are: signature, gait, and keystroke. Some traits could be considered as mixture of physiological and behavioral characteristics, e.g., voice.

There are three main categories of biometric applications: verification, identification, and surveillance. Verification systems verify the authenticity of a claimed identity based on the input biometric sample. Identification systems determine if the input sample is associated with any of a large number of enrolled identities. Surveillance systems determine whether a person belongs to a watch list of identities.

A biometric verification system is considered as a signal detection system with pattern recognition architecture. Hence it consists of the following modules: a signal sensor that senses the raw biometric, a signal processor that extracts some informative set of features from the raw signal, and a classifier that compares features against a biometric model or some templates stored in system database. In the enrollment phase, some biometric samples are acquired by a sensor then quality of samples is checked. If a sample passes the quality test, features are extracted and stored as a template in system database, or they are used to develop a biometric model for the enrolled person. In the authentication phase, a query sample is acquired and used for feature extraction, then it is matched with stored templates or model of the claimed person.

##### **1.1.1 Performance of biometric systems**

Quality of a biometric system is represented by its accuracy of recognition, security and complexity. Accuracy of a biometric system is the ability to detect genuine signals and discriminate

forgeries. Some measures for recognition accuracy are the false reject rate (FRR) and the false accept rate (FAR). FRR is the percentage of the genuine samples rejected by the system. FAR is the percentage of impostor samples accepted by the system. Different than simple passwords, biometric passwords do not provide a perfect recognition tool, because of the intra-personal variability and the inter-personal similarity of biometric traits. Fuzziness of biometrics might result from imperfect signal acquisition, e.g., unaligned fingerprints, or it might be an intrinsic feature of the biometric traits, e.g., variations in the handwritten signatures. The later is harder to cancel and a trade-off between FRR and FAR exists.

Receiver operating characteristics (ROC) curve is a visual tool that permits compromising this trade-off. Classifier outputs (scores) are stored and used for ROC curve generation. A point on the ROC curve represents a compromise between genuine accept rate ( $GAR = 1 - FRR$ ) and FAR, when a specific score is used as a classifier threshold. Therefore, FAR for a specific threshold is the ratio of forgery samples with a score higher than this threshold. GAR is the ratio of genuine samples with a score higher than the threshold. Area under the curve (AUC) reflects the recognition power of the classifier, as high AUC implies possibility to achieve high GAR and low FAR simultaneously.

Although biometrics provides a trusted mean of authenticity, it might involve security vulnerabilities (Uludag, 2006). One source of attacks is bypassing the system with fake biometric samples. For instance, dummy fingers or forged signatures might be used. This security issue can be alleviated by increasing the classifier threshold, so fake samples are rejected. Another issue is related to security of the biometric templates. In case that templates are stored for verification, they can be copied from the system. Once compromised, the biometric sample might be used to access multiple accounts of the user. In addition, compromised traits might be irrevocable. For instance, a user has to change his handwritten signature if it is compromised. A counter measure against such attacks is to enforce template protection through applying the cancelable biometrics approach (Ratha *et al.*, 2001). Templates are stored in a transformed form so that they are more secure. Once compromised, a different transformation function can be applied to same trait, so it is revocable. Finally, a biometric system can be attacked by

insiders through overriding classifier functionality. For instance, outputs of a biometric system can be overridden so it produces predetermined decision labels. This type of attacks might be prevented by employing the so-called bio-cryptography approach (Uludag *et al.*, 2004). Instead of generating traditional classification labels, cryptographic keys are produced through a relatively complex process that is hard to overridden. A released key can be considered as a positive classification label, or it could be further employed to execute an encryption or a digital signature cryptographic scheme. For the aforementioned biometric security approaches, a sample is classified in a transformed domain, which impacts recognition accuracy.

Besides accuracy and security of a biometric system, its complexity is an important aspect. Reliable systems should be simple in terms of operational memory and processing time. There is a trade-off between system complexity and its accuracy and security. For instance, for unstable behavioral biometrics, like handwritten signatures, high dimensional feature representations are needed to capture variation of signatures. In addition, with numerous users that have high inter-personal similarities, user clusters are split by complex decision boundaries. Therefore, complex classifiers must be designed to model these boundaries and provide acceptable recognition accuracy. Moreover, enforcing security of biometric systems involves overhead on the resource required for operation.

### **1.1.2 Handwritten signature biometrics**

Handwritten signature is the biometric trait the most universally employed for authentication. Designing verification systems based on this behavioral biometric trait is challenging, as it involves high intra-personal variability and inter-personal similarity. Signature Verification (SV) systems are either online or offline systems (Impedovo and Pirlo, 2008). In online SV, dynamic signals, e.g., pressure, velocity, stroke order, etc., are acquired during the signing process by special pens and tablets. On the other hand, OLSV systems rely on static features acquired from the signatures images scanned after the signing process. Since physical presence of signer is not mandatory for the offline systems, they can be employed in a broader range for

applications. However, the offline static features are less stable and easier to forge than the online features.

Signature forgeries can be classified in three main categories: random, simple, and simulated forgeries. For random forgery, the forger produces the signature randomly, as neither he has access to a signature template nor he knows the name of the signer. For simple forgery, the forger has no access to the signature but he knows the name, so he produces signatures based on the name. For simulated forgery, the forger has access to a signature sample, and he simulates the genuine signature.

Similar to the other biometric verification systems, SV systems consist of signature acquisition, feature extraction, and classification modules. Generally, signature features can be classified into two main categories: global and local. Global features concern with the whole signature. Typical global features are number of components, global orientation of the signature, envelopes, coefficients obtained by mathematical transforms, etc. Local features are extracted from specific parts of the signature. Depending on the level of details captured, local features can be divided into component oriented features which are extracted at the level of each component, e.g., height to width ratio of the stroke, relative positions of the strokes, stroke orientation, etc., and pixel-oriented features, which are extracted at pixel level, e.g., grid-based information, pixel density, graylevel intensity, texture, etc. It is worth noting that some parameters, which are generally considered to be global features, can also be applied locally, and vice versa. For instance, contour-based features can be extracted at the global level, e.g., envelopes of the whole signature, or at the local level, e.g., envelopes of each connected component.

## **1.2 Bio-cryptography**

Cryptography is the study of mathematical techniques related to aspects of information security such as confidentiality, data integrity, and authenticity (A. Menezes, 1996). Different cryptographic schemes has been applied to achieve such security goals as data encryption and digital signatures. Most of cryptographic techniques rely on keys for operation. There are two main categories: symmetric-key and asymmetric-key cryptosystems. Symmetric-key algorithms use

trivially related, often identical, cryptographic keys. On the other hand, asymmetric-key cryptography uses different keys for encryption and decryption. The encryption keys are public and they are used by all users to encrypt data, while each user has his own private key by which he decrypts the data.

The drawback of cryptography lies in its dependency on secret cryptographic keys, that if compromised, security of the system is compromised. Although the cryptographic keys are too long to be guessed by impostors, they are also too long to be memorized by the legitimate users. This problem is alleviated through storing the key in a secure place, e.g., a smart card, and a user retrieves his key by providing a simple password. Such token/password solution forms a weak point in a security system, as whatever strong is the cryptographic key, overall system security is determined by the password length. Moreover, these authentication measures are not strongly associated with the user identity, so they cannot really distinguish between attackers and legitimate users. Any person who steals the password and the card can access the system.

To alleviate this key management problem, biometric passwords, e.g., fingerprint, iris, handwritings, etc., replace the traditional passwords within the so-called bio-cryptographic schemes (Uludag *et al.*, 2004). For instance, a cryptographic key is locked by a handwritten signature image and a user must provide a genuine signature sample to unlock his cryptographic key. Once unlocked, the key can be used to decrypt user confidential data or it is used to sign his information digitally. As biometrics are strongly associated with user identity, and it is less likely that they are stolen or forgotten, so they guarantee authenticity of the cryptography system users. However, design of the bio-cryptography systems is challenging due to the fuzzy nature of the biometric traits. The intra-personal variability and inter-personal similarity of biometric signals lead to false rejection of authorized users and acceptance of unauthorized users, respectively.

### **1.2.1 Bio-cryptographic schemes**

The main bio-cryptographic schemes are: key release, key generation, and key binding (Jain *et al.*, 2006). In the key release mode, both cryptographic keys and biometric passwords are

stored separately in system database. A key is released to its owner through a traditional biometric verification process. This scheme is not secure, as both the cryptographic and biometric information are stored in a plain form and they can be stolen or edited. In key-generation schemes, the biometric trait is used to generate the cryptographic key directly through some transformation functions. It is not easy to generate strong, robust, and random cryptographic keys from variable and correlated biometric signals.

The key-binding scheme is the most reliable bio-cryptographic scheme. Both the cryptographic key and the biometric template are combined in a single template in a way that it is impossible to decouple these two parts without knowing the decoupling scheme, and providing a genuine biometric sample (Soutar *et al.*, 1999). In these schemes, classical crypto-keys are used. Therefore, keys are as strong, random, accurate, and unique as with the classical cryptographic systems. However, the inter-personal similarity and the intra-personal variability of biometrics might lead to false acceptance of impostors or false rejections of genuine users, respectively. The Fuzzy Vault (FV) scheme is the most commonly employed key-binding scheme that alleviates this fuzziness problem (Juels and Sudan, 2002).

### 1.2.2 Fuzzy Vault scheme

The FV is a cryptographic construction that binds a secret message, e.g., cryptographic key, with an unordered and/or fuzzy locking set (Juels and Sudan, 2002). In the authentication time, the secret message can be decoupled if the unlocking set substantially matches the locking set. Accordingly, the FV construction can be used efficiently to secure cryptographic keys by using fuzzy and unordered features extracted from the biometric traits, e.g., minutiae in fingerprints, cross points in signatures, etc, as locking/unlocking sets.

A FV scheme locks a cryptographic key  $K$  by means of a biometric template  $T$ . To unlock  $K$ , a biometric query sample  $Q$  is provided by the user. Figure 1.1 illustrates this locking/unlocking process. For key locking,  $K$  is split into  $k + 1$  strings and constitutes a coefficient vector  $C = \{c_0, c_1, c_2, \dots, c_k\}$ . A polynomial  $p$  of degree  $k$  is encoded using  $C$ , where  $p(x) = c_k x^k + c_{k-1} x^{k-1} + \dots + c_1 x + c_0$ . Then, a locking set  $F^T = \{f_i^T\}_{i=1}^t$  is extracted from  $T$ . The



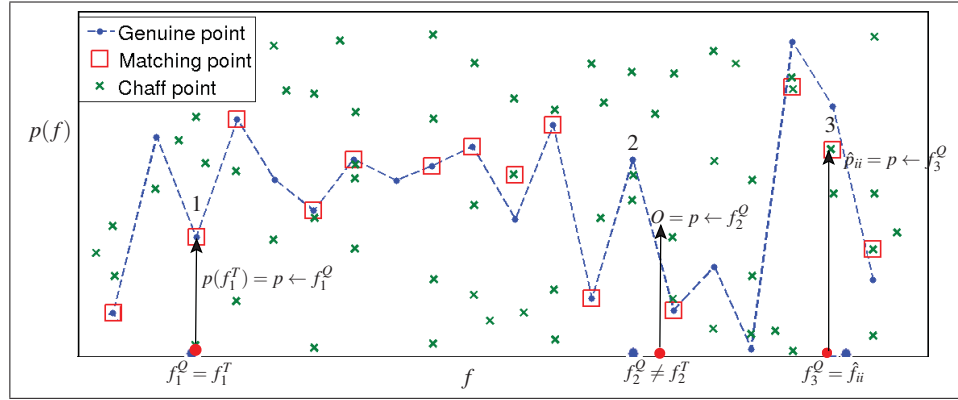


Figure 1.1 Illustration of the FV locking/unlocking process.

polynomial is evaluated for all points in  $F^T$  and constitutes the set  $p(F^T) = \{p(f_i^T)\}_{i=1}^t$ . The points  $(F^T, p(F^T))$  constitutes the genuine vault points.

It is known that, for a polynomial of degree  $k$ , only  $k + 1$  points on its curve are needed to reconstruct the polynomial equation. So, the genuine vault points can be used to reconstruct the polynomial  $p$ , and thereby the cryptographic key  $K$ . Hence, any person who accesses the genuine points can retrieve the key. Accordingly, to conceal these data from attackers, a set of  $z$  chaff (noise) points  $(\hat{F} = \{\hat{f}_{ii}\}_{ii=1}^z, \hat{P} = \{\hat{p}_{ii}\}_{ii=1}^z)$  are generated. Then, the chaff and genuine points are mixed to constitute the vault  $V_T$  of length  $r$  points. Security of the vault relies on the amount of concealing chaffs. In case that an impostor accesses the vault data, he has to search for at least  $k + 1$  genuine points, out of  $r = t + z$  points of the FV. This search task becomes infeasible with high number of chaff points  $z$ .

The proper way to unlock  $K$  from the vault  $V_T$ , by legitimate users, is to apply a biometric query sample  $Q$ . An unlocking set  $F^Q = \{f_j^Q\}_{j=1}^t$  is extracted from  $Q$ . Then, the chaff points are filtered by matching items of  $F^Q$  against all items in  $V_T$ . In the ideal case, each feature encoded in  $F^Q$  locates the corresponding genuine feature encoded in  $F^T$  (e.g., point 1 in Figure 1.1). On the other hand, due to the fuzzy nature of biometrics, some elements of  $F^Q$  differ from their corresponding elements in  $F^T$ , and two types of errors might occur, namely erasures and noise. For the erasures case,  $f_i^Q$  does not match with any vault point, so it does not add any element to the matching set (e.g., point 2 in Figure 1.1). For the noise case, a feature

$f_i^Q$  might equate a chaff  $\hat{f}_{ii}$ , so that it adds a noise point  $(\hat{f}_{ii}, \hat{p}_{ii})$  to the matching set (e.g., point 3 in Figure 1.1).

Finally, the resulting matching set is fed to a polynomial reconstruction algorithm, to reconstruct the encoded polynomial  $p$ . This process succeeds only if the matching set contains at least  $k + 1$  genuine points. However, even if enough genuine points exists, it is not possible to differentiate between the genuine and noise points. To overcome this, FV decoders employ error correction codes, like Reed-Solomon (R-S) codes. The genuine set  $(F^T, p(F^T))$  is considered as a code word of length  $t$ , that encodes a secret message of length  $k + 1$ , where there are  $t - k - 1$  redundancy elements. During the decoding process, some noise is added to this code producing a corrupted version of it. The error correction codes can correct some of these errors and recover the secret message.

#### 1.2.2.1 State of the art of FV

In literature, the FV design problem is addressed with different approaches. Generally, authors proposed methodologies to absorb dissimilarities between template and query biometric signals, so that they are within the error correction capacity of the decoder. Researchers applied the FV scheme using different biometric traits. For instance, Clancy (Clancy *et al.*, 2003) and Yang (Yang and Verbauwhede, 2005) proposed FV systems based on fingerprints and studied impact of FV parameters on system performance. The later replaced the Cartesian coordinates by the polar coordinates for aligning fingerprint queries and templates before matching. Also, they used multiple templates for FV encoding.

Uludag (Uludag *et al.*, 2005) proposed an error detection approach for fingerprint-based FVs, where CRC-codes replace the Reed-Solomon (RS) correction codes for polynomial reconstruction. In addition, a methodology for chaff points generation is proposed, where chaffs are separated from locking points by a specific distance threshold. Therefore, the probability that chaffs interfere with the genuine points decreases. Moreover, the authors proposed a fingerprint alignment technique using helper data.

Nandakumar (Nandakumar *et al.*, 2007) proposed a FV system based on fingerprints. The contributions of this work are: i) filtering most of chaff points by a coarse filter, ii) usage of the minutiae point orientation as an extra discriminating feature, iii) quantization of features adaptively based on the noise level of the signal, and vi) usage of multiple queries for authentication.

Y. Wang (Wang and Plataniotis, 2007) proposed a FV system based on the face images. Features variability is alleviated through quantization of distances vectors instead of the absolute feature values. Lee (Lee *et al.*, 2008) proposed a FV system based on the iris traits and used the polar representation for alignment. Features are extracted from multiple region of interest (ROI) of the iris. A group of ROI from multiple iris templates constitute a batch. Batches are represented in a high dimensional features space. Using k-means, centers of different clusters are identified and mapped to fixed feature codes used to encode the FV.

Freire (Freire-Santos *et al.*, 2006) and (Freire-Santos *et al.*, 2007) investigated applicability of FV implementation based on the handwritten signatures. While the online signatures demonstrated acceptable FV accuracy, the authors observed that static features extracted from the offline signature images involve too much variability to build reliable FV systems.

Some authors proposed modified FV schemes that utilize passwords for an additional security layer (K.Nandakumar *et al.*, 2007) and (Reddy and I.Babu, 2008). In the encoding phase, the feature vector is encrypted using a random function derived from user password. Then, the transformed features vector generates the vault points. At last, the constituted vault is encrypted by a key derived from the password. If the password is compromised, system security degrades to the original FV security level. Also, this approach alleviates the FAR without having much impact on the FRR.

Some authors proposed FV systems based on multiple biometric traits, where performance of multi-modal FV systems outperformed the single modal systems (Nagar *et al.*, 2011), (Nandakumar, 2008), (Hirschbichler, 2008), (Meenakshi, 2010).



## CHAPTER 2

### OPTIMIZED DISSIMILARITY REPRESENTATIONS WITH APPLICATION TO SIGNATURE VERIFICATION AND BIO-CRYPTOGRAPHY

The dissimilarity representation (DR) provides a classification space that is defined by some proximity measure. One case where the DR approach is advantageous is when patterns are represented in high-dimensional feature spaces, and only simple classification rules are applicable. For instance, bio-cryptographic schemes use biometric signals to secure cryptographic keys. These schemes mostly employ an error correction code that is considered as a simple threshold classifier. In addition, for behavioral biometrics, e.g., handwritten signatures, effective verification systems rely on high-dimensional feature representations and complex classifiers. It is a challenge to produce discriminant bio-cryptographic implementations based on behavioral biometrics, with these limitations on representation size and classification complexity. In this chapter, an approach is proposed for optimization of DRs, so that a concise representation is discriminant even by employing a simple threshold classifier. To this end, high-dimensional feature representations are translated to an intermediate space, where pairwise feature distances are the space constituents. Then, Boosting Feature Selection algorithm is applied in this intermediate space, and produces an adaptive dissimilarity measure that relies on a concise feature representation. This measure generates the final dissimilarity space, where pattern proximities to some prototypes are the space constituents. Finally, discriminant prototypes are selected in the dissimilarity space for enhanced representation. The proposed approach is applied to classical and bio-cryptographic systems for offline signature verification. Proof of concept simulations on the Brazilian signature database indicate the viability of the proposed approach. Concise DRs with only 20 features and a single prototype are produced. With employing a simple threshold classifier, the produced DRs have shown state-of-the-art accuracy of about 7% average error rate, as that of complex systems in the literature. The content of this chapter was published at the 2<sup>nd</sup> International workshop on Similarity-Based Pattern Analysis and Recognition (Eskander *et al.*, 2013f), the 2<sup>nd</sup> International workshop on Automated Forensic Handwriting Analysis (Eskander *et al.*, 2013a), and submitted to the spe-

cial issue of the IEEE Transactions on Neural Networks and Learning Systems on "Learning in non-(geo)metric spaces" (Eskander *et al.*, 2013e).

## 2.1 Introduction

Traditional classification techniques employ feature representations (FRs), where discriminative features are extracted from patterns that can be viewed as points in the feature space. An effective FR implies that patterns belong to same class are close, while patterns of different classes are well separated in the feature space. In case with numerous classes, with high dimensional representations, and with a limited number of labeled patterns per class, the design of an efficient FR becomes infeasible (Guyon and Elisseeff, 2003).

Recently, the concept of dissimilarity representation (DR) has been introduced (Pekalska and Duin, 2002). Instead of designing classifiers in the feature space, proximity (similarity/dissimilarity) measures define the classification space. The rational behind this approach is that modeling the proximity between patterns may be more discriminative than modeling the objects themselves. Indeed, objects belonging to a specific class have a shared degree of commonality that can be captured by dissimilarity measure. To this end, proximity measures are computed and considered as features for classification. These measures can be derived in many ways, e.g. from raw (sensor) measurements, histograms, strings or graphs. However, it can also be build on top of a FR space (Duin *et al.*, 2010).

Defining a DR by measuring proximities between patterns is beneficial when designing a classifier in the original FR space is intractable (Pekalska and Duin, 2005). For instance, for applications with patterns represented by high-dimensional FRs, and that with complex decision boundaries, complex nonlinear classifiers must be designed. By exploiting a DR space, proximities between high dimensional patterns are reduced to patterns in a lower dimensional space, where simple linear classifiers may be sufficient.

Efficiency of a DR depends on the extend to which the proximity measure reflects the real proximities between patterns. In case of a feature-based DR, a feature vector is extracted

from both the questioned and prototype samples, and a certain distance function measures the proximity between the two feature vectors. Designing a proximity measure implies selection of suitable features and a distance function. Moreover, the class prototypes should be selected to be robust and discriminant (Pekalska *et al.*, 2006).

This chapter proposes an approach for optimizing DRs through selection of representative features, distance functions, and prototypes. A global DR is initially optimized based on training data from a set of classes, and then tuned for new classes using their specific data.

The advantages of the proposed approach are as follows:

The DR is designed so that the within-class (WC) and between-class (BC) dissimilarities constitute two compact and separated clusters. Accordingly, the problem is shifted from representing objects of known classes to representing the relation between any two objects from similar or dissimilar classes. As we model the relation between classes, not the classes themselves, so designed representations are considered as class-independent representation (CIR). The CIR could be further tuned to specific classes to produce class-specific representations (CSR). This property might facilitate designing global systems that can be tuned to specific classes. For instance, biometric systems, that rely on physiological or behavioral human traits for authentication, e.g., fingerprint, face, iris, signatures, etc., are designed based on samples from users who are already enrolled prior to the design phase (Jain *et al.*, 2006). Classifiers that are designed in the original FR space produce systems that are tuned to users who provide training samples. These systems might not generalize well for new users, who are enrolled after designing the system. The proposed approach can be employed to design global biometric classifiers that generalize for future users.

In addition, the proposed approach absorbs some of the intra-class dissimilarities, and increases the inter-class dissimilarities. Hence, the WC and BC distributions are easily separable in the DR space, and simple classification rules, e.g., thresholds, could be sufficient.

Finally, although the ultimate goal of the approach is to design reliable DRs, where the dissimilarity-based classifiers can be developed, the proposed approach can also be employed to design tradi-

tional feature-based classifiers. For instance, it could be used as a tool for feature selection, where the curse of dimensionality is alleviated through shifting the multi-class problem, with few samples per class, to a more tractable two-class problem with large number of class samples. This way, the features embedded in the designed DR are filtered and feature-based classifiers are designed in the resulting feature space. Moreover, performance of some feature-based distance classifiers, e.g., KNN., rely on the employed distance functions and prototypes, and intensive research focused on this area (Ramanan and Baker, 2011),(Garcia *et al.*, 2012). The distance functions and prototypes embedded in the designed DR can support the design of such distance classifiers.

Two applications are considered for proof-of-concept in this chapter: offline signature verification (OLSV) (Impedovo and Pirlo, 2008) and Bio-cryptography (Uludag *et al.*, 2004). OLSV systems verify that a signature image belongs to a specific writer. Design of these systems is challenging, as signature images are represented by high-dimensional vectors and there is no knowledge on either the forgeries nor signatures of future users during the design phase. Recently, Rivard *et al.*, proposed a writer-independent (WI) OLSV system that generalizes for unseen users and forgeries (Rivard *et al.*, 2013). The approach proposed in this chapter extends this system, where here we generalize this approach for designing class-independent representations, that can be further tuned to new classes. Basic concepts of the proposed approach has been appeared in (Eskander *et al.*, 2013a). Also, a detailed implementation of employing the proposed approach as an intermediate tool for feature selection, where final classifiers are designed in the resulting FR spaces, has been appeared in (Eskander *et al.*, 2013c). Here, we implement the overall DR optimization methodology, in order to investigate the discriminative power of the optimized DRs. To this end, accuracy of a simple classifier (just a threshold), that uses the resulting concise DRs (based on few features and a single prototype), is compared to complex systems in the literature.

Bio-cryptographic system design is another application considered in this chapter. In these systems, biometric signals lock the cryptographic keys within security schemes like encryption and digital signatures (Uludag *et al.*, 2004). The idea behind these schemes is to consider



the query biometric signal as a noisy version of its prototype. If the query sample is genuine, the dissimilarity between the query and its prototype is limited, so this noise can be eliminated and the locked cryptographic key is released to its owner. This problem is also challenging as while biometric signals might be represented with high-dimensional vectors, such systems need that only few features are used to lock the cryptographic key. In addition, the error correction decoders embedded in such systems can be considered as simple classification thresholds, and therefore no complex classification rules can be employed. We employ the proposed approach to design representations that are adapted to this application. A preliminary version of this approach has been appeared in (Eskander *et al.*, 2013f), and a complete bio-cryptographic implementation has been appeared in (Eskander *et al.*, 2014a). Here, we show that the design of a bio-cryptographic system can be formulated as a classical classifier in the dissimilarity (D-space). Accordingly, same design approach is applied for both verification and bio-cryptographic systems. Also, we show that applying the prototype selection method, provides a way for enhancing accuracy and complexity of these systems.

For proof of concept simulation, the Brazilian signature verification DB is used (Freitas *et al.*, 2000). Representations are optimized based on the proposed approach, where the effect of each processing step on the representation effectiveness is measured by its impact on separating WC and BC samples. The resulting representations are used to design signature based systems, and the classification error rates are reported.

The rest of the chapter is organized as follows. In the next section, some related works for dissimilarity learning are discussed. Section III presents the proposed dissimilarity representation optimization approach. The application on OLSV and bio-cryptography are presented in sections IV and V respectively. Finally, the experimental methodology and some research results are presented and discussed in section VI.

## 2.2 From learning features to the dissimilarity representation

Designing a classifier relies somewhat on the concept of dissimilarity. Ideally, similar objects should produce similar classification labels and dissimilar objects should produce dissimilar

labels. Similarity learning takes place either implicitly or explicitly, based on the applied representation and learning strategy. In this section, we discuss these different forms and their relation to the proposed approach.

### 2.2.1 Learning feature representations

The approach to independently design a FR, as a pre-processing step for the classifier design, is known as filter feature selection approach (Guyon and Elisseeff, 2003). As the compactness and isolation of different class distributions implies that real dissimilarities between objects are captured, some methods rely on these measures to guide the feature selection process. For instance, the Fisher criterion is extensively employed, where the ratio between WC and BC variance reflects the effectiveness of the representation (Fisher, 1936). This approach, mostly, involves optimization problems that becomes infeasible, when large number of classes are represented by few training samples and high dimensional feature extractions.

Alternatively, some methods, namely, wrapper and embedded feature selection, combine the design of both representation and classifier in a single process. For the wrapper approach, a fixed predictor is tested based on different candidate representations, where the minimum classification error determines the best representation (Kohavi and John, 1997). For the embedded approach, the predictor is built and tuned concurrently with selection of an effective representation. Examples under this category are classification and regression trees (CART) (Breiman *et al.*, 1984), and boosted feature selection (BFS) (Tieu and Viola, 2004), where individual features are selected in a greedy manner, while building the classifier. Although they make searching in high dimensional spaces more tractable, these methods do not produce generic representations, as they tune the representation to specific classification rules. Moreover, feature selection techniques do not necessarily produce FRs that generalize for unseen classes and samples during operations.

### 2.2.2 Learning distance functions

A more explicit way for dissimilarity learning is done with classifiers that take explicit distances (or kernels) as inputs, e.g., KNN, SVM, etc. For such distance/kernel-based classifiers, a distance function that measures the true proximity between FRs of patterns are firstly designed, then they are fed to the classification stage. Performance of such classifiers relies on the quality of the resulting proximity measure, which in turn relies on the employed FR, the distance function applied to the representation, and the prototypes that used as references for distance computations.

In the literature, such systems are optimized through employing distance function learning (Ramanan and Baker, 2011), and/or prototype selection (Garcia *et al.*, 2012). Distance function learning is done through optimizing a parametrized function, so the WC distances are minimized and BC distances are maximized. Examples of the employed distance functions are  $L_2$  distance (Frome *et al.*, 2007), Chi-squared (Domeniconi *et al.*, 2002), wighted similarity (Babenko *et al.*, 2009), probability of belongingness to different classes (Mahamud and Hebert, 2009). However, most of employed distance functions take the following form:

$$D_Q(x, x_i) = (x_i - x)^T Q (x_i - x). \quad (2.1)$$

where,  $x_i$  and  $x$  are the FR, for the questioned and the prototype samples, respectively. This technique provides a way to translate hardly separable distributions to a space where the distributions are more separable. In order that conventional pattern recognition approaches holds in the new space,  $Q$  is restricted to be a symmetric and positive definite matrix (or kernel), so that  $D$  is a metric function (Ramanan and Baker, 2011). According to Eq. 2.1, it is obvious that entries of the  $Q$  matrix determine the impact of the pairwise distances between individual features on the proximity measure. So, learning  $Q$  implies feature selection. It is shown that accuracy of this metric increases, when full matrices are considered (i.e., not only a diagonal matrix but some weighted relations among individual distances exist) (Bar-Hillel *et al.*, 2005). However, this comes with the expense of increased complexity of the optimization problems.

Also, it is shown that global distance functions does not frequently represent all classes (Weinberger and Saul, 2009). Instead, the concept of local distance functions is presented (Ramanan and Baker, 2011). For instance, the metric tensor concept is represented, where instead to learn a metric  $Q$  for the whole space, specific metric  $QT$  is learned for every class  $T$ . This approach becomes complex for large number of classes, and some authors suggested grouping similar classes under larger classes, so trade-off between global and class specific similarity functions can be achieved (Babenko *et al.*, 2009). Moreover, as aforementioned, quality of the proximity measure depends also on the prototype set used as a reference for distance measuring. Prototype selection is extensively studied for distance-based classifiers like KNN (Garcia *et al.*, 2012).

### 2.2.3 Learning dissimilarity representations

Recently, the concept of distance function learning is generalized to learning dissimilarity representations (DRs) (Pekalska and Duin, 2002). While distance functions are restricted to be feature-based and metric, conversely, these conditions are relaxed in the DRs, so any proximity measure can be employed. Through this, the statistical pattern recognition methods are applicable to subjects indescribable by traditional feature representation and/or that involve none metric proximities. Previously, such subjects could only be classified through structural pattern recognition methods, hence, the DR approach is considered as a bridge between structural and statistical pattern recognition techniques. In addition, the DR approach can also be applied to feature-based systems, where the learning and classification tasks are more tractable in the DR space than in the FR space. This last scenario relies on same reasoning like that of the kernel trick, while, here, any proximity measure can be employed (Pekalska and Duin, 2005).

To produce a DR, firstly, the pairwise distances constitute a dissimilarity matrix, where a row of this matrix represents distances of questions samples to all prototypes. The prototypes may include the whole training set, however, some approaches are applied to reduce the prototype set for the most informative ones (Pekalska *et al.*, 2006). Then, the dissimilarity matrix can

be translated to the so-called dissimilarity (D-space) through two main techniques, namely, embedding and vectorial representation (Pekalska and Duin, 2005).

In the embedding technique, the dissimilarities are embedded in a new space where the dissimilarities are preserved. The transformation function should guarantee that the new space is Euclidean, so classical pattern recognition theories hold their. For the vectorial representation technique, the dissimilarities are used to produce a vectorial space directly, where distances to the prototypes are the space dimensions. So, a row in the dissimilarity matrix is a vector in this space (Pekalska and Duin, 2006). Designing classifiers in such space models the relations between pattern proximities. Instead of applying local rules to a dissimilarity cell, like KNN, more globally aware rules, that consider the other rows of the dissimilarity matrix, are applied. It is found that simple classifiers applied in the dissimilarity space, perform even better than complex classifiers that run in the original feature space (Pekalska and Duin, 2006).

#### **2.2.4 Proposed approach**

The proposed approach seeks, not to design DR-based classifiers, but to learn the DR itself. This way, simple classification rules, e.g., thresholds, might be sufficient when employed in the DR space, or even if they are directly applied to the dissimilarity matrix.

In this Thesis, we are concerned only with the feature-based DRs, and vectorial D-spaces. As the proposed approach involves feature selection, it can also be considered as a tool to design pure feature-based classifiers. Moreover, as it involves distance function learning and prototype selection, the approach can be employed to design distance-based classifiers that work in the FR space. Compared to the formal distance function defined in Eq. 2.1, our approach learns a simple identity matrix, so only unweighted feature selection is achieved, and no cross-feature relations are modeled. Through this simplified metric, the learning process is lighter but it might not be optimal. To compensate against such lack of information, an adaptive dissimilarity measure is proposed, that absorbs some of the intrinsic feature variability.

Most important, the produced DRs are designed so that a global class-independent representation represents all, and even unseen, classes, where it could be tuned for specific classes by means of class-specific training data. This strategy is analogue to the global and local distance concepts applied for the distance-based classifiers, and the proposed approach could be employed in such context. Also, this property might facilitate the design of adaptive systems, that are designed with limited data, then they adapt whenever new data becomes available, e.g., adaptive biometric systems.

### 2.3 Dissimilarity Representation Optimization approach

In this section, the proposed approach for optimizing DRs is illustrated. Although the DR is a general approach, where dissimilarity measures can be derived directly from patterns, e.g., sensor measurements, we discuss here the special case where the DR is build on top of a feature representation (FR). This approach is suitable for applications, where many techniques of feature extraction are already proposed.

Figure 2.1 illustrates a DR constituted on top of a FR. Assume a system is designed for  $U$  different classes, where for any class  $u$  there are  $R$  prototypes (templates)  $\{p_{ur}\}_{r=1}^R$ . Also, a class  $v$  provides a set of  $J$  questioned samples  $\{Q_{vj}\}_{j=1}^J$ . The dissimilarity between a questioned sample  $Q_{vj}$  and a prototype  $p_{ur}$  is  $\delta^{Q_{vj}p_{ur}}$ . Dissimilarities between all the questioned and prototypes samples constitute a dissimilarity matrix, where each row contains distances from a specific query to all of the prototypes.

In case that questioned and prototype samples belong to the same class, i.e.,  $u = v$ , the dissimilarity sample is a WC sample (black cells in Figure 2.1). On the other hand, if questioned and prototype samples belong to different classes, i.e.,  $u \neq v$ , then the dissimilarity sample is a BC sample (white cells in Figure 2.1).

Effective DR implies that all of the WC distances have zero values, while all of the BC distances have large values. This occurs when the employed dissimilarity measure absorbs all of the WC variabilities, and detects all of the BC similarities. The proposed approach aims to enlarge

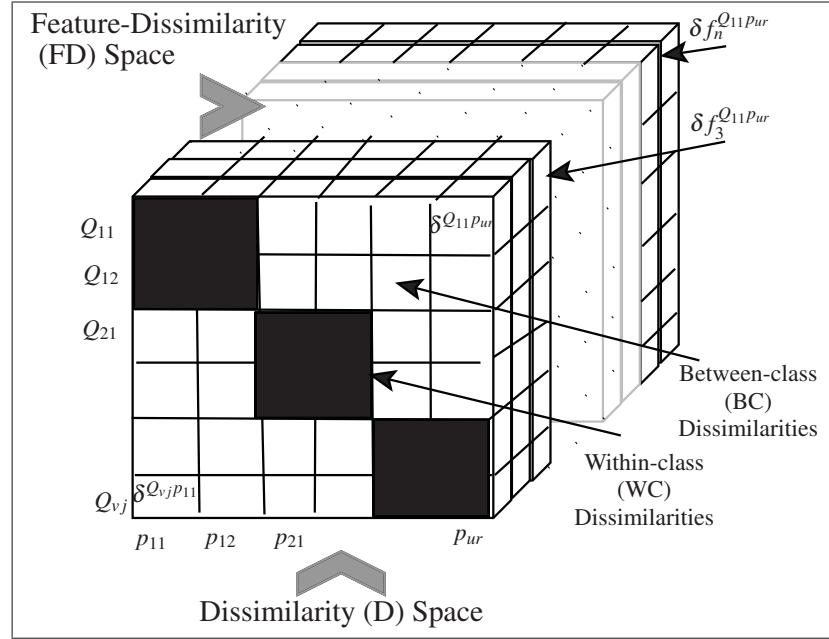


Figure 2.1 Illustration of a dissimilarity representation (DR) built on top of a feature representation (FR): black and white cells represent WC and BC dissimilarities, respectively. The third dimension represents the feature dissimilarity (FD) space, where dissimilarities between prototype and query samples are measured by the distance between their feature representations. The dissimilarity cells constitute a dissimilarity matrix, where each row contains the distances from a specific query to all of the prototypes.

the separation between the BC and WC distributions, so that simple classification rules can be applied in the resulting dissimilarity space, or even applied directly to the dissimilarity matrix.

### 2.3.1 Feature selection and dissimilarity learning

As shown in Figure 2.1, the third dimension represents the feature dissimilarity (FD) space, where dissimilarities between a query  $Q^{vj}$  and a prototype  $p^{ur}$  are measured by the dissimilarity between their feature representations  $\{f_n^{Q^{vj}}\}_{n=1}^N$  and  $\{f_n^{p^{ur}}\}_{n=1}^N$ , respectively. These representations are translated to a FD space of same dimensionality  $N$ , and constitute a dissimilarity vector:

$$d^{Q_{vj}p_{ur}} = \{\delta f_n^{Q_{vj}p_{ur}}\}_{n=1}^N, \quad \delta f_n^{Q_{vj}p_{ur}} = \|f_n^{Q_{vj}} - f_n^{p_{ur}}\| \quad (2.2)$$

Distance of a dissimilarity vector is measured through employing a distance function  $g$ , to determine the value of corresponding dissimilarity matrix cell:

$$\delta^{Q_{vj}p_{ur}} = g(\{\delta f_n^{Q_{vj}p_{ur}}\}_{n=1}^N) \quad (2.3)$$

It is obvious that, independently on the distance function  $g$ , discriminative power of the employed FD representation controls the separability of the WC and BC cells. However, proper selection of the function  $g$  is also important, to capture the maximum power of the underlying representation. Accordingly, we first select features that discriminate between the WC and BC samples in the FD space, then a proper distance function is designed.

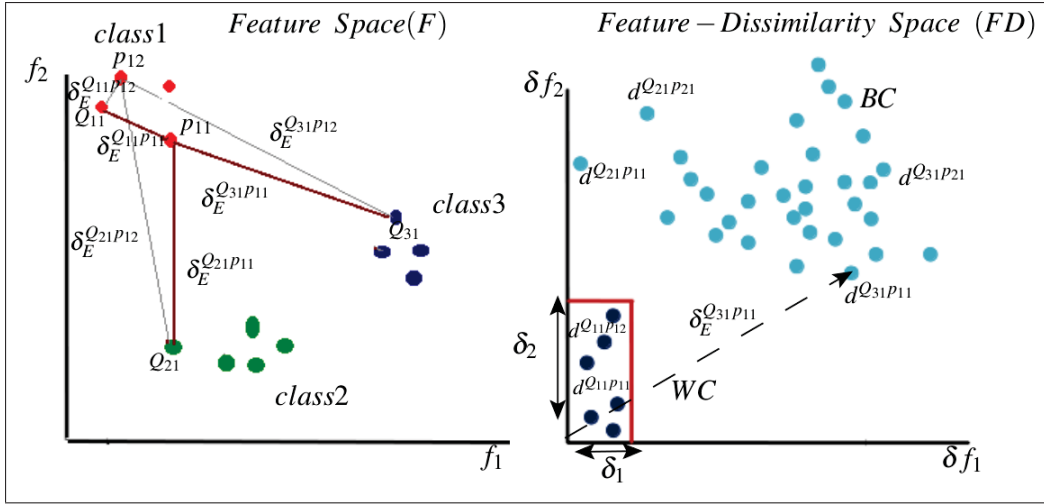


Figure 2.2 Illustration of feature selection in the original feature space  $F$  (left) and in the feature-dissimilarity space  $FD$  (right). Distance between two samples in the  $F$  space is translated to a vector in the  $FD$  space, where each dimension represents the distance as measured by a single feature.

To illustrate the proposed feature selection approach, consider the example shown in Figure 2.2. In the left side objects from three classes are represented in the FR space  $F$ . For simplicity, only two features  $f_1$  and  $f_2$  are shown in this figure, while typical representations might have high dimensionality. In this example, we assume that class 1 has two prototypes  $p_{11}$  and  $p_{12}$ . Also, consider, for now, that the employed distance function  $g$  is the Euclidean distance:



$$\delta_E^{Q_{vj}P_{ur}} = \sqrt{\sum_{n=1}^N (\delta f_n^{Q_{vj}P_{ur}})^2} \quad (2.4)$$

It is clear that a DR that is built on top of this FR is discriminative. WC distances (like  $\delta_E^{Q_{11}P_{11}}$ ) are generally smaller than the BC distances (like  $\delta_E^{Q_{21}P_{11}}$ ). However, in the  $F$  space, the impact of each feature on the WC and BC distances is not clear. With representations of high dimensionality, high number of classes, a small number of training samples per class, it is not feasible to select the most discriminative features in the feature space  $F$ .

On the other hand, in the FD space, the impact of every individual feature on the WC and BC distances is clear, (see right side of Figure 2.1). In this space, a distance  $\delta_E^{Q_{vj}P_{ur}}$  is represented by the distance from the origin point to  $d^{Q_{vj}P_{ur}}$  (as defined in Eq. 2.2 and Eq. 2.4). Accordingly, projecting the dissimilarity vector on different axis of the FD space, determines the discriminative power of each dimension. For instance, it is obvious that  $\delta f_2$  is more discriminative than  $\delta f_1$ . For all samples belonging to class 1, (like  $Q_{11}$ ),  $\delta f_2^{Q_{11}P_{1r}} < \delta_2$  and for all other class samples (like  $Q_{21}$  and  $Q_{31}$ ),  $\delta f_2^{Q_{vj}P_{1j}} > \delta_2$ . On the other hand,  $\delta f_1$  is less discriminant. For class 2 query  $Q_{21}$ ,  $\delta f_1^{Q_{21}P_{11}} < \delta_1$ , same as that for class 1 sample  $Q_{11}$ . Besides that it is easier to rank features in the  $FD$  space, the multi-class problem with few training samples per class in  $F$  space is transformed to a two-class problem in the new space, with more training samples per class.

It is also important to note that, proper ranking of dissimilarity features in the FD space leads to proper ranking of the features in the original  $F$  space. For instance, the BC sample  $d^{Q_{21}P_{11}}$  is correctly classified by the dissimilarity feature  $\delta f_2$  (see right side of Figure 2.2), while it is misclassified as a WC dissimilarity by  $\delta f_1$ . Similarly, while  $f_2$  easily splits the different clusters in the  $F$  space (see left side of Figure 2.2),  $f_1$  hardly splits class 1 and class 2 as  $f_1^{P_{11}}$  is very close to  $f_1^{Q_{21}}$ . This property enables employing the FD space as a tool for designing reliable FRs, for traditional feature based classifiers. FRs are designed in the FD space, then samples are represented and classified in the resulting feature space.

### 2.3.2 A two-step feature selection approach

The aforementioned feature selection methodology can be applied to optimize a dissimilarity matrix that includes pairwise distances relating to all classes. Such approach has some drawbacks:

- The resulting representation might fit only the classes used for training, and might not generalize well for unseen classes.
- The resulting representation, and in turn the designed proximity measure, are designed to fit all of the classes, i.e., global measures. This is not easily achievable, and it is shown that local distance functions outperform the global ones (Ramanan and Baker, 2011). To illustrate that, see Figure 2.1 (left side). While  $f_1$  is not discriminant enough to separate class 1 from the other 2 classes, and  $f_2$  is better for this task,  $f_1$  is more suitable to separate class 3 from the other classes. In order to design class specific solutions, one may suggest that one-against-all dissimilarity matrix is optimized for each class. However, the WC entries for each matrix will be few, so searching in high dimensional representations will be intractable.

To avoid the above drawbacks, and achieving effective representations/proximity measures that generalize for unseen classes, we propose a two-step feature selection approach as shown in Figure 2.3. In the first step, a class-independent database (CID) containing samples from a set of classes, is used for training. These classes should represent the other unseen classes. For example, in case of designing OLSV systems, these classes (users) have their objects (hand-written signatures) written with same alphabet, collected under same acquisition conditions, etc. This way a measure that discriminates between WC and BC dissimilarities extracted from the training samples, will be discriminant for other classes that are not used for training (Rivard *et al.*, 2013). To this end, high dimensional feature extractions from the CID samples are trans-

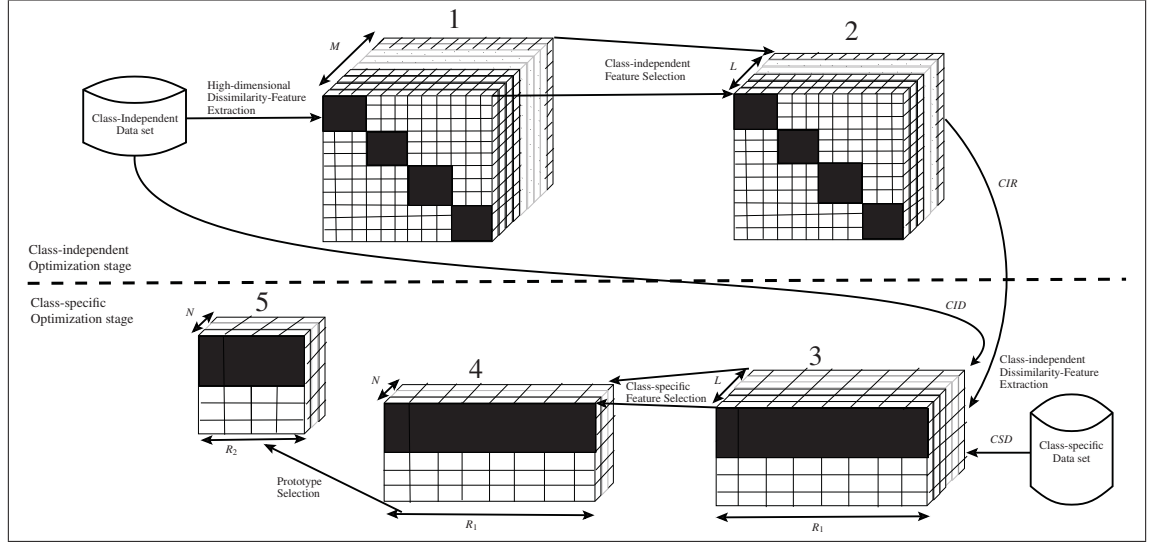


Figure 2.3 A framework of the optimizing dissimilarity representations approach, consists of a) class-independent optimization phase and b) class-specific optimization phase. a) High dimensional FRs are extracted from a class-independent dataset (CID), and translated to the FD space of same dimensionality  $M$ . The composed dissimilarity matrix (1) are not optimized and the WC and BC entries are not separable. A class-independent feature selection process runs in the FD space, producing a class-independent representation (CIR) of reduced dimensionality  $L$ . The resulting reduced dissimilarity matrix (2) should have more separable clusters. b) whenever class-specific data (CSD) exist,  $R_1$  samples from this data, and some samples from the CID, are represented in the CIR of dimensionality  $L$ . Then, a class-specific dissimilarity matrix (3) is composed, where WC entries consist of dissimilarities within the  $R_1$  class prototypes, and the BC entries consist of dissimilarities between samples from the CID and the  $R_1$  prototypes. This matrix (3) still has its clusters not well separated as the embedded FR is not tuned to the specific class. A second feature selection process runs in the reduced FD space, producing a class-specific representation (CSR) of reduced dimensionality  $N$ . The resulting matrix (4) might have their clusters more separable, as the CSR is further tuned. However, not all columns of (4) have same separability. For enhanced representation, entries of matrix (4) are translated to a dissimilarity (D-space), where the  $R_1$  prototypes are the space dimensions. Then, a feature selection process runs in the D-space, with class prototypes are considered as features, and the best  $R_2$  prototypes are selected. This way, a CSR of  $N$  features and  $R_2$  prototypes might result in separable WC and BC clusters of the class-specific matrix (5).

lated to the FD space, and constitute WC and BC samples of same dimensionality  $M$ . Feature selection is done in the FD space, producing a CIR of  $L < M$  dimensionality. This CIR can be used to represent samples from unseen classes, and might be used to define global proximity measure with acceptable generalization accuracy.

In the second step, whenever enough samples from a specific class exist, they are used to tune the CIR and produces a CSR that fits that class. To this end, a class specific database (CSD) containing  $R1$  samples, besides some samples from the CID that represent the other classes, are used for training. All samples are represented in the CIR space defined through the first step, and they are translated to the FD though computing dissimilarities between all samples. As the search space has a reduced dimensionality  $L$ , so training with few class-specific samples becomes more tractable in this reduced space. Then, additional feature selection step runs in this space, producing a class specific representation (CSR) of dimensionality  $N < L < M$ .

The two feature selection steps can be achieved by employing different feature selection methods in the FD space. However in this Thesis, this concept is realized by employing the boosting feature selection (BFS) method (Tieu and Viola, 2004), for fast searching in high dimensional spaces. Decision-stumps ( $DS$ ) (Iba and Langley, 1992), that are single-split single-level classification trees, are trained through a boosting process (Schapire, 2002). Training of a  $DS$  is equivalent to selection of a single feature that discriminates between two classes based on a splitting threshold. If the BFS runs in the  $FD$  space, a  $DS_i$  at a learning iteration  $i$ , locates the best dissimilarity feature  $\delta f_i$ , that splits the two classes around a splitting dissimilarity threshold  $\delta_i$ .

### 2.3.3 Adaptive dissimilarity measure

Selecting the most discriminant features in the  $FD$  space, produces representations with low WC and high BC dissimilarities. However, some of the intrinsic fuzziness of the samples are not canceled through this feature selection process. To illustrate this, see Figure 2.4 (left side). Although the BC and WC clusters are separated, samples of each class are scattered in the space, because of the intra-class variability and inter-class similarities. To alleviate this variability, we propose an adaptive dissimilarity measure. This measure is computed in the  $FD$  space, and absorbs some of the intrinsic fuzziness. For a feature representation  $F = \{f_n\}_{n=1}^N$ , the feature dissimilarity vector  $\Delta = \{\delta_n\}_{n=1}^N$  is learned in  $FD$  space, where  $\delta_n$  discriminates between the WC and the BC dissimilarities for a feature  $f_n$ . Based on this modeled dissimilar-

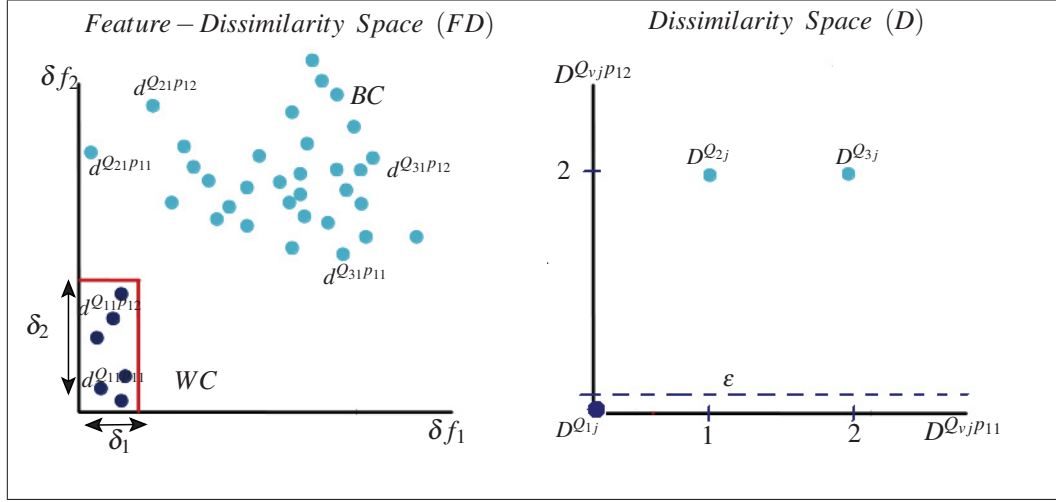


Figure 2.4 Illustration of the transformation from the feature-dissimilarity space (left) to the dissimilarity space (right). All dissimilarity samples that belong to same query  $Q_{vj}$  in the FD space, are represented as a single vector  $D^{Q_{vj}}$  in the D-space. Dimensions of this vector are distance components to individual prototypes. In case that only the best prototype is selected, employing a simple threshold  $\epsilon$  might be enough to discriminate between WC and BC samples in the D-space.

ity, we replace the Euclidean distance measure ( $\delta_E^{Q_{vj}P_{ur}}$ ) (defined in Eq. 2.4), by the adaptive dissimilarity measure:

$$\delta_A^{Q_{vj}P_{ur}} = \sum_{n=1}^N (\delta_n^{Q_{vj}P_{ur}}), \quad \delta_n^{Q_{vj}P_{ur}} = \begin{cases} 0 & \text{if } (\delta_n^{Q_{vj}P_{ur}} < \delta_n) \\ 1 & \text{otherwise} \end{cases} \quad (2.5)$$

Employing this measure absorbs some of the intrinsic feature variability and increases its discriminative power. For instance, according to Eq. 2.5, distances among the WC samples  $\delta_A^{Q_{vj}P_{ur}} = 0, \forall v = u$ . Moreover, for most of the BC distances  $\delta_A^{Q_{vj}P_{ur}} = 2, \forall v \neq u$ .

### 2.3.4 Prototype selection in the dissimilarity space

The aforementioned steps might be utilized to produce class specific FRs or distance measures, for designing traditional or distance-based classifiers in the feature space, respectively. Also, designed dissimilarity matrices are expected to have well separated WC and BC cells, so simple rules, e.g., KNN, might be directly applied to the matrix.

However, the ultimate goal of the proposed approach is to generate a reliable DR space, in which more complex classifiers can be designed. Efficiency of a DR depends on the discriminative power of its generating prototypes. Accordingly, a prototype selection method is proposed. In this Thesis we employed the vectorial representation to generate a dissimilarity (D)-space, where distances to the prototypes are the space dimensions. A feature selection process runs in the D-space, to locate the most discriminating dimensions.

Figure 2.4 illustrates the transformation between the  $FD$  and  $D$  spaces. In the left side, WC and BC samples are represented in the  $FD$  space. It is obvious that different prototypes (different columns in the dissimilarity matrix), produce different distance values, where significant variability exists for the WC and the BC classes. Also, in this space, it is not clear which prototype is the most informative. In the right side, samples are projected to a vectorial dissimilarity class-specific D-space, where distances to the prototypes constitute its dimensions (for simplicity only two prototypes are shown for class 1, however, practical D-spaces could be of higher dimensions). Hence, each row in the dissimilarity matrix is presented as a vector in the D space, and each column represents projections of all vectors on a specific prototype.

Consider, for a class  $u$ , the available set of  $R_1$  prototypes  $P = \{p_{u1}, p_{u2}, \dots, p_{uR_1}\}$ . The adaptive dissimilarity distance for a query  $Q_{vj}$  is computed against every prototype  $p_{ur} \in P$ , according to Eq.2.5. This operation produces a dissimilarity vector  $D^{Q_{vj}}$  in the D-space, where

$$D^{Q_{vj}} = \{D^{Q_{vj}p_{u1}}, D^{Q_{vj}p_{u2}}, \dots, D^{Q_{vj}p_{uR_1}}\}. \quad (2.6)$$

It is clear that the D-space provides easier way to rank prototypes according to their discriminative power. For instance, for class 1,  $p_{12}$  is more discriminative than  $p_{11}$ , as for all BC samples,  $D^{Q_{vj}p_{12}} = 2$ . While for  $Q_{21}$ ,  $D^{Q_{21}p_{11}} = 1$  (because  $\delta f_1^{Q_{21}p_{11}} < \delta_1$ ). So, for this class, measuring the dissimilarity relative to  $p_{12}$  results in more isolated clusters, and a smaller threshold, e.g.  $\epsilon = 1$ , can discriminate between the WC and BC dissimilarities. The prototype selection method can be realized by various feature selection techniques (with considering prototypes as features), however, we realized it through employing the BFS approach (Tieu and Viola, 2004).

## 2.4 Application to signature verification

### 2.4.1 Dissimilarity-based signature verification

Here, we employ the optimized DRs to design OLSV systems, where handwritten signature images are used for authentication. Different than online SV systems, where signature dynamics such as velocity, pressure, etc., are acquired during the signing process, the OLSV systems rely on static features extracted from signature images, producing a harder pattern recognition problem (Impedovo and Pirlo, 2008).

Standard OLSV systems are designed in the FR space. The training samples should represent a wide range of genuine signatures and possible forgeries, for all system users. For real world applications, e.g., banking systems, the number of users could be very high and there is a high risk of forgery. The enrolling signature samples, available for designing such systems, are mostly few and no samples of forgeries are available. In addition, high-dimensional feature representations are needed to capture the invariance of the signature images. With these limitations, it is a challenge to design FRs, that absorb the intra-personal variabilities while detecting both the forgeries and the inter-personal similarities. Also, feature-based OLSV systems are writer-dependent, as they are designed for some specific writer(s), and they do not necessarily generalize for unseen classes.

Recently, the DR concept is applied to the OLSV problem, to alleviate the limitations of the FR-based design approach. Siteargur N. Srihari et al., (Srihari *et al.*, 2004) developed generative models from the WC and BC statistics, where the correlation between binary features is used as a distance measure. The WC distribution is modeled based on either writer-specific samples or all-writers samples, resulting in writer-dependent (WD) and writer-independent (WI) OLSV systems, respectively.

Concurrently, Santos, Bertolini, and Sabourin et al., proposed similar DR based systems (C. Santos *et al.*, 2004)-(Bertolini *et al.*, 2010), where the Euclidean distance between graphometric feature vectors is used as a distance measure. However, they developed discriminative WI

classifiers that learn from independent (population) database, and no samples from real system users are incorporated in the design phase. The designed systems are tested with signatures from users who are not seen during the design phase, and show acceptable accuracy. The proved hypothesis in this work is: if a huge number of samples are used to build a global DR-based classifier, it is statistically valid that the resulting model generalizes for users whose samples are not included in the training set.

For the above systems, no representation optimization is applied, only fixed feature representation, and in turn fixed distance measure, is employed. Later, Batista et al., (Batista *et al.*, 2010) applied the dissimilarity learning concept to produce reliable WD-SV systems. A feature-based one-class classifier is built by producing user-specific generative models using Hidden Markov models (HMMs). These models are considered as prototypes, where the likelihood that a query signature belongs to the different models constitute a D-space, and a discriminative classifier is designed in this space. This implementation implies optimized DRs, as the designed HMMs are the DR constituents.

Recently, Rivard et al., (Rivard *et al.*, 2013) extended the work in (Bertolini *et al.*, 2010), and achieved higher performance, when employing an embedded feature selection methodology through BFS over huge number of multi-scale multi-type features. In this case, the DR is optimized concurrently with building the classifier. This work is the base of our proposed approach, where only the class-independent optimization stage is implemented, producing a CIR. This CIR is embedded in a global classifier designed in the class-independent FD space. Here, we extend Rivard approach, by employing the class-specific optimization stage shown in Figure 2.3.

#### **2.4.2 Applying the proposed approach**

The user-specific training signature samples from the CSD, and some samples from the CID dataset, are represented in the CIR feature space. Then, these FRs are translated to a class-independent FD space, where a class-specific feature selection process runs. A D-space is constituted by computing the adaptive dissimilarity measures for the different samples against



the user prototypes. Finally, the best prototype is selected in the D-space, by running a BFS process with the prototypes are considered as features. Compared to Rivard system, the proposed SV implementation employ concise user-specific unweighted FR. Also, in this chapter, we restricted our implementation for single dimensional D-spaces, where only the best prototype is incorporated in the verification decision. Accordingly, the SV system functionality is given by:

$$SV(Q_{vj}, p_{ur*}) = \text{sign}(\epsilon - \delta_A^{Q_{vj}p_{ur*}}). \quad (2.7)$$

where  $Q_{vj}$  is a query signature  $j$  of user  $v$ ,  $p_{ur*}$  is the best prototype (with index  $r^*$ ) selected for user  $u$ ,  $\epsilon$  is a dissimilarity threshold defined in the D-space, and  $\delta_A^{Q_{vj}p_{ur*}}$  is the user-specific adaptive dissimilarity measure defined by Eq. 2.5.

## 2.5 Application to bio-cryptography

### 2.5.1 Dissimilarity-based bio-cryptography

Bio-cryptography is another example where the proposed approach is beneficial. Bio-cryptographic systems are introduced to replace the traditional usage of simple user passwords by biometric traits like fingerprint, iris, face, signatures, etc., to secure the cryptographic keys within security schemes like encryption and digital signatures (Uludag *et al.*, 2004).

Robust bio-cryptographic systems operate in the key-binding mode where classical crypto-keys are coupled with a biometric message. For key binding, some encoding schemes like Fuzzy Commitment (Juels and Wattenberg, 1999) and Fuzzy Vault (FV) (Juels and Sudan, 2002) are the most commonly employed. In the enrollment phase, a prototype biometric message encodes the secret key. In the authentication phase, a message is extracted from the query sample to decode the key. If the query sample is genuine, the dissimilarity between the encoding and decoding messages is limited, so this dissimilarity can be eliminated by the decoder. On the other hand, if the query sample belongs to another person, or if it is a forged sample, the dissimilarity between the two messages is too high to cancel. Accordingly, the secret key will be

unlocked only to users who apply similar enough query samples. Practical decoding complexity of such codes needs that employed biometric messages should be concise. It is a challenging task to produce a concise and informative messages from the biometric signals, and to use simple classifiers like the bio-cryptographic decoders to differentiate between genuine and forged samples.<sup>1</sup>

In literature, the concept of dissimilarity representation is not directly employed to design bio-cryptographic systems. However, some authors proposed methodologies to absorb the dissimilarities between encoding and decoding biometric signals, so that they are within the error correction capacity of the decoder. For instance, fingerprint-based fuzzy vaults are designed by using some minutia points extracted in the spatial space to constitute the FV locking features. Dissimilarity between locking and unlocking features is decreased by aligning query and template fingerprints, and by applying an adaptive bounding box during matching the minutia points (Nandakumar *et al.*, 2007).

### 2.5.2 Applying the proposed approach

In this Thesis, we consider the FV key binding cryptographic scheme. In FVs, a locking message  $F^{p_{ur}} = \{f_n^{p_{ur}}\}_{n=1}^N$  is extracted from a biometric enrolled signal  $p_{ur}$  of user  $u$ , and it locks the user cryptographic key  $K$ . To conceal this locking messages from attacker, a set of chaff (noise) points are mixed with the locking elements.

In the authentication phase, a user provides a biometric query signal  $Q_{vj}$ , that produces an unlocking message  $F^{Q_{vj}} = \{f_n^{Q_{vj}}\}_{n=1}^N$ . Each unlocking element  $f_n^{Q_{vj}}$  is matched against all of the locking elements of  $F^{p_{ur}}$ , and producing a matching set. The key  $K$  can be unlocked only if the error of the matching set is beyond the FV error correction capability  $\epsilon$ .<sup>2</sup> There are two sources of matching errors: erasures and noise. In the erasures case, some unlocking elements do not match their corresponding locking elements, so they are not added to the matching set.

---

<sup>1</sup>Details of how the crypto-key is encoded/decoded by means of biometrics is out of the scope of this chapter. More details on these aspects are discussed in Chapter 4.

<sup>2</sup>Error correction capacity  $\epsilon$  of a FV bio-cryptographic systems relies on the sizes of both the cryptographic key and the encoding messages. Also, for technical issues, the message elements  $\{f_i\}_{i=1}^N$  are quantized in 8-bit words before computing the dissimilarities. See Chapter 4. for detailed explanation.

In the noise case, some unlocking elements match some of the chaff points, so they are added as noise  $\delta'$  to the matching set. For efficient FV implementation, sum of these errors should not exceed  $\epsilon$  for genuine query signals, while it exceeds it for impostors.

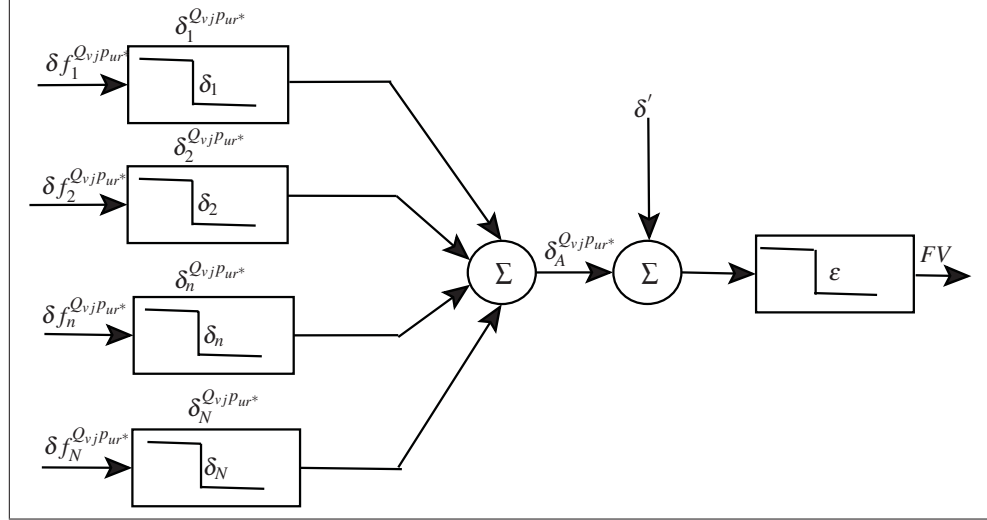


Figure 2.5 Proposed model of the FV functionality: every unlocking element  $f_n^{Q_{vj}}$  is matched against all locking elements  $\{f_n^{P_{ur*}}\}_{n=1}^N$ , where it succeeds to locate the corresponding element only if their dissimilarity is within the modeled dissimilarity threshold  $\delta_n$ . To correctly decode the FV, the overall dissimilarity between the locking and unlocking messages  $\delta_A^{Q_{vj}P_{ur*}}$ , besides the noise error  $\delta'$  (resulting from false matching with chaffs), should not exceed the error correction capacity  $\epsilon$  of the decoder.

Based on this FV description, we employ the proposed DR optimization approach, so the FV functionality can be modeled as shown in Figure 2.5. A FV unlocking element  $f_n^{Q_{vj}}$  is matched against all locking elements with a tolerance  $\delta_n$ . So, a locking element is successfully located when the corresponding unlocking elements is similar enough (has limited dissimilarity  $\delta_n$ ). Accordingly, the dissimilarity results from the erasures error type is equivalent to  $\delta_A^{Q_{vj}P_{ur*}}$  (see equation 2.5), where  $r^*$  is the prototype selected for key locking. Considering the extra noise errors  $\delta'$ , the total error should not exceed the FV error correction capacity  $\epsilon$ . Accordingly, the proposed formulation of the FV functionality is:

$$FV(Q_{vj}, p_{ur*}) = \text{sign}(\epsilon - (\delta_A^{Q_{vj}P_{ur*}} + \delta')). \quad (2.8)$$

where  $FV = 1$  implies that the locked cryptographic key is released (that should occur only if  $u = v$ ), and  $FV = 0$  implies that the key cannot be released,  $Q_{vj}$  is a query signature  $j$  of user  $v$ ,  $p_{ur*}$  is the best prototype (with index  $r^*$ ) selected for user  $u$ ,  $\epsilon$  is a FV error correction capacity (and it is equivalent to the dissimilarity threshold defined in the D-space), and  $\delta_A^{Q_{vj}p_{ur*}}$  is the user-specific adaptive dissimilarity measure defined by Eq. 2.5.

A complete implementation of a FV system based on offline signature images has been appeared in (Eskander *et al.*, 2014a). However, here, the concept is consolidated with emphasizing on the DR-based formulation of the FV design problem. According to Eq. 2.7 and Eq. 2.8, It is obvious that, if no chaff points are incorrectly matched with unlocking elements, i.e., noise error  $\delta' = 0$ , then functionalities of both the cryptographic and classical SV systems are similar. Hence, same design approach and steps can be followed to build both systems. Moreover, we investigated the impact of selecting effective prototypes for FV encoding on the system accuracy and complexity.

## 2.6 Experiments

Classification of handwritten signature images is a suitable example, where the power of proposed approach can be investigated. The static signals extracted from such images involve high variability between intra-personal signatures, and similarities between inter-personal signatures. Also, it is easy to imitate the signature images by forgeries. Generally, such hard classification problem is tackled though employing high dimensional representations and complex classifiers (Impedovo and Pirlo, 2008). Here, we test the optimized DRs for distinguishing between genuine and forged signatures, where concise representations and the simplest classification rule (threshold) are employed.

### 2.6.1 Database

The Brazilian database is used for proof-of-concept simulations (Freitas *et al.*, 2000). It contains 7,920 samples of signatures that were digitized as 8-bit grayscale images over 400X1000 pixels at resolution of 300 dpi. This DB contains genuine signatures, simple and simulated

forgeries. For simple forgery, the forger knows the writer's name but not the signature morphology, so he writes the name using his style of writing. A simulated forgery has access to a sample of the signature, and he imitates its image. To generate random forgeries for a specific user, where a forger is not supposed to know neither the name nor the signature morphology, we consider genuine signatures of other users as forgeries.

The signatures were provided by 168 writers. For the last 108 writers, there are only 40 genuine signatures per writer and no forgeries. We consider these signatures as the CID, and they are employed for the class-independent optimization phase. For the first 60 writers, there are 40 genuine signatures, 10 simple forgeries and 10 simulated forgeries per writer. These signatures are considered as CSD, and used for the class-specific optimization stage and for performance evaluation.

## 2.6.2 Class-independent optimization

The processing steps included in the class-independent optimization stage are executed as shown in the top part of Figure 2.3. This stage aims to produce a CIR of relatively low dimension  $L$ , from rich feature extractions of huge dimensionality  $M \gg L$ . Same experimental settings are applied, as that employed by Rivard et al., as this work is considered as the base of our approach (Rivard *et al.*, 2013). The relatively low dimensionality of the produced CIR, besides it represents unseen users, it facilitates designing of further tuned CSRs of concise representations, where limited CSDs are available.

### 2.6.2.1 Feature extraction

Extended-shadow-code (ESC) (Sabourin and Genest, 1994), and directional probability density function (DPDF) (Drouhard *et al.*, 1996) are employed. Features are extracted based on different grid scales, hence a range of details are detected in the signature image. A set of 30 grid scales is used for each feature type, producing 60 different single scale feature representations. These representations are then fused to produce a FR of huge dimensionality,  $M = 30,201$  (Rivard *et al.*, 2013).

Table 2.1 The class-independent data set (CID):  $U^{CID} = 108$  users  $\times$   $R_1 = 40$  genuine signatures. Only some of unique dissimilarities are taken from the huge dissimilarity matrix, and used to design the CIR.

| <b>Training set</b><br>( $R_1^T = 30$ signatures/user) |   | <b>Validation set</b><br>( $R_1^V = 10$ signatures/users) |  |
|--|---|---|--|
| WC   | BC  | WC  | BC   |
| distances among $R_1^T$ signatures                     | distances among 29 signatures from $R_1^T$ and 15 signatures of other users | distances among the $R_1^V$ and $R_1^T$                   | distances among $R_1^V$ and 30 signatures selected randomly from other users |
| 108x30x29/2<br>=46,980 samples                         | 108x29x15<br>=46,980 samples  | 108x10x30<br>=32,400 samples                              | 108x10x30<br>=32,400 samples   |

### 2.6.2.2 Class-independent feature selection

The initial dissimilarity matrix (see matrix 1 in Figure 2.3), is constituted by translating the FR, of  $M$  dimensionality, for  $U^{CID} = 108$  users of the CID and  $R = 40$  prototypes per writer, to a FD space of same dimensionality. To this end, dissimilarities among all signatures ( $108$  writers  $\times$   $40$  signatures) constitute a dissimilarity matrix, as shown in Figure 2.1. As this matrix is huge, not all of its cells are used to design the CIR. Also, to avoid overfitting the CID data, some of the WC and BC samples are used for training and other set of samples are used for validation. Table 2.1 illustrate the selection of these datasets, where all the data and experimental settings are the same as found in (Rivard *et al.*, 2013).

Finally, a BFS process is executed in the FD space, producing a CSR of reduced dimensionality  $L = 555$ . It is possible to use this representation, and the corresponding learned dissimilarity thresholds  $\{\delta_l\}_{l=1}^L$ , for designing a global (class-independent) adaptive dissimilarity measure as defined in Eq. 2.5. However, in case of having enough data for a specific class, it is used to tune the global model to a CSR and a class-specific dissimilarity measure, by employing the following optimization stage.

Table 2.2 The class-specific datasets (CSDs): a different CSD for each of the 60 users, with 60 signatures each: 40 genuine+10 simple +10 simulated forgeries. 30 genuine signatures are used as prototypes, so  $U^{CSD} = 1 \times R_1^{cs} = 30$  genuine signatures. The other samples are used as queries for performance evaluation, so  $30 \times 40 = 1200$  dissimilarities are used for testing.

| <b>Training set</b><br>( $R_1^{cs} = 30$ genuine signatures) |   | <b>Testing set</b><br>(10 genuine signatures+<br>10 simple+10 simulated forgeries) |  |
|--|---|--|--|
| WC   | BC  | WC   | BC   |
| distances among<br>the $R_1^{cs}$<br>signatures              | distances among<br>29 signatures<br>from $R_1^{cs}$<br>and 15 signatures<br>from all users of CID | distances among<br>the 10 genuine<br>signatures and<br>$R_1^{cs}$                  | distances among 30<br>forgeries (10 simple<br>+10 simulated<br>+10 random selected<br>from other users) and $R_1^{cs}$ |
| 30x29/2<br>=435 samples                                      | 29x108x15<br>=46,980 sample   | 10x30<br>=300 samples  | 30x30<br>=900 samples  |

### 2.6.3 Class-specific optimization

The processing steps included in the class-specific optimization stage are executed as shown in the lower part of Figure 2.3. This stage aims to produce a CSR of concise dimension  $N < L \ll M$ , by tuning the CIR to the specific user. This representation is used to design a user-specific dissimilarity measure. Also, a vectorial D-space is designed using the user prototypes, where best prototypes can be selected.

#### 2.6.3.1 Class-specific feature selection

The CSD, and some samples from CID, are used to generate a dissimilarity matrix (see matrix 3 in Figure2.3). Only  $R_1^{cs} = 30$  signatures are considered as prototypes. The WC cells are generated by considering unique dissimilarities among these prototypes, where dissimilarities are computed in a CIR-FD space of dimensionality  $L = 555$ . To generate BC samples, dissimilarities between  $R_1^{cs}$  and some signatures selected from all users of the CID are computed. Details of the WC and BC training data is illustrated in Table 2.2.

Then, a BFS process runs in this reduced CIR space, where it is found that only 40 boosting iterations result in saturated performance. To unify the representation size for all users, we selected the best  $N = 20$  features as a CSR. This representation, and the corresponding learned dissimilarity thresholds  $\{\delta_n\}_{n=1}^N$ , are employed to design a class specific adaptive dissimilarity measure as defined in Eq. 2.5. The dissimilarity cells are re-computed based on this class specific solution and produced a dissimilarity matrix, where the WC and BC are more separable (see matrix 4 in Figure 2.3).

### 2.6.3.2 Prototype selection in the D-space

Finally, the dissimilarity matrix is used to produce a D-space of dimensionality  $R_1 = 30$ , where each row is represented as a vector in this space. A BFS runs in this space to rank the prototypes. Here, we chose only the best prototype to generate a single dimensional concise DR, where  $R_2 = 1$  and  $N = 20$  (see matrix 5 in Figure 2.3). This DR is used for performance evaluation, where the testing dataset (illustrated in table 2.2), is used to generate a testing dissimilarity matrix based on this representation.

### 2.6.4 Performance evaluation

The proposed approach is evaluated by investigating its power to generate well separated WC and BC dissimilarities. A testing dissimilarity matrix is generated for each of the 60 users of the CDs, as illustrated in Table 2.2. For each user, dissimilarities between the  $R_1^{cs} = 30$  prototypes are computed against 40 query samples. Of these: 10 genuine signatures, 10 simple, 10 simulated, and 10 random forgeries are employed and result in 300 WC and 900 BC samples.

To investigate the different processing steps of the proposed approach (shown in figure 2.3), the impact of each step, on separating the WC and BC clusters, are decoupled through employing the following experiments:

- without feature selection:  $N = 20$  features are randomly selected from the  $M$  feature extractions.



- class-independent feature selection: the best  $N$  features, from  $L$  features of the CIR, are used. This setting investigates to which extend does a CIR generalize to unseen users.
- class-specific feature selection: the CSR, of dimensionality  $N$ , is used. For this, and for the above experiments, we tested the representation power of the features by employing the following strict dissimilarity measure:

$$\delta_S^{Q_{vj}P_{ur}} = \sum_{n=1}^N (\delta_n^{Q_{vj}P_{ur}}), \quad \delta_n^{Q_{vj}P_{ur}} = \begin{cases} 0 & \text{if } (\delta_{f_n}^{Q_{vj}P_{ur}} = 0) \\ 1 & \text{otherwise} \end{cases} \quad (2.9)$$

this way, we decouple the impact of the adaptive dissimilarity measure, and we only test the applicability of the CIR to adapt for specific users.

- class-specific feature selection with adaptive dissimilarity measure: the CSR is used, where the adaptive dissimilarity measure (defined in Eq. 2.5) is employed. So, the impact of absorbing feature variability, through employing the adaptive dissimilarity measure, is tested in this experiment.

For all above cases, the testing dataset (shown in Table 2.2) is used to generate dissimilarity matrices, according to the investigated representation. Then, separability of the WC and BC clusters are measured by the Hellinger distance (Cha, 2007). Assuming normal distributions of the WC and BC clusters. The squared Hellinger distance between them is give by:

$$H^2(WC, BC) = 1 - \sqrt{\frac{2\sigma_1\sigma_2}{\sigma_1^2 + \sigma_2^2}} e^{-\frac{1}{4} \frac{(\mu_1 - \mu_2)^2}{\sigma_1^2 + \sigma_2^2}}. \quad (2.10)$$

where,  $\mu_1, \mu_2$  and  $\sigma_1, \sigma_2$  are the mean and variance values for  $WC$  and  $BC$ , respectively.

To measure the clusters separability for the different types of forgeries, we compute  $H_{random}$ ,  $H_{simple}$  and  $H_{simulated}$ , where the parameters  $\mu$  and  $\sigma$  of the BC cluster are computed each time, based on the dissimilarities against samples of a specific type of forgeries. Also, we report  $H_{all}$ , where the distribution parameters are computed according to dissimilarities of all forgery types. For all cases, these measures are averaged over all users, e.g.,  $\hat{H} = \frac{\sum_{u=1}^{U^{cs}} H_u}{U^{cs}}$ .

According to equations 2.7 and 2.8, accuracy of both classical and bio-cryptographic signature verification systems, could be considered similar in case that the noise error  $\delta'$  is canceled. In our experiments, we considered the case of zero noise error, so that both systems have same recognition rates.<sup>4</sup> As the recognition accuracy relies on the dissimilarity ranges separability and on the employed dissimilarity threshold  $\epsilon$  (see Eq. 2.7 and Eq. 2.8), we measure the recognition errors for all of the dissimilarity scores and use them to generate ROC curves. A ROC curve plots the false accept rate (FAR) against the genuine accept rate (GAR) for all possible thresholds (all generated dissimilarity scores). FAR for a specific threshold is the ratio of forgery samples with a dissimilarity score smaller than this threshold. GAR is the ratio of genuine samples with a dissimilarity score smaller than the threshold.

In order to have a global assessment on the quality of the SV and bio-cryptographic systems, that are built on the optimized DRs, we compute and average the area under the ROC curves (AUC), for all users in the *CSD* subset. High AUC indicates more separation between the dissimilarity score distributions for the genuine and impostor classes.

In order to measure actual recognition rates, we set the dissimilarity to  $\epsilon = 6$ , this value is empirically selected to compensate between the FRR and FAR errors<sup>3</sup>.

To assess the impact of the prototype selection step, we compare the recognition rates for cases where prototype selection is employed or not. For both cases, we report the average error rate ( $AER_{all}$ ), where

$$AER_{all} = (FRR + FAR_{random} + FAR_{simple} + FAR_{simulated})/4 \quad (2.11)$$

---

<sup>4</sup>Here we considered the case when no chaff points are embedded, so impact of the noise error is neglected. This scenario is achievable when we generate chaffs adaptively based on the learned feature dissimilarities. Proof of concept of such adaptive chaff generation appeared in (Eskander *et al.*, 2013b), where high number of chaffs are embedded with minimal impact on the recognition rate. For more details see Appendix II.

<sup>3</sup> $\epsilon = 6$  is equivalent to encoding a crypto-key of 128 – bits by a biometric message of length  $N = 20$ , by implementing the FV key-binding scheme (Juels and Sudan, 2002)

False reject rate ( $FRR$ ) is the ratio of genuine queries, that produce dissimilarity scores above the threshold  $\epsilon$ ,  $FAR_{random}$ ,  $FAR_{simple}$  and  $FAR_{simulated}$  are the ratio of random, simple, and simulated forgeries, respectively, that produce dissimilarity scores less than  $\epsilon$ .

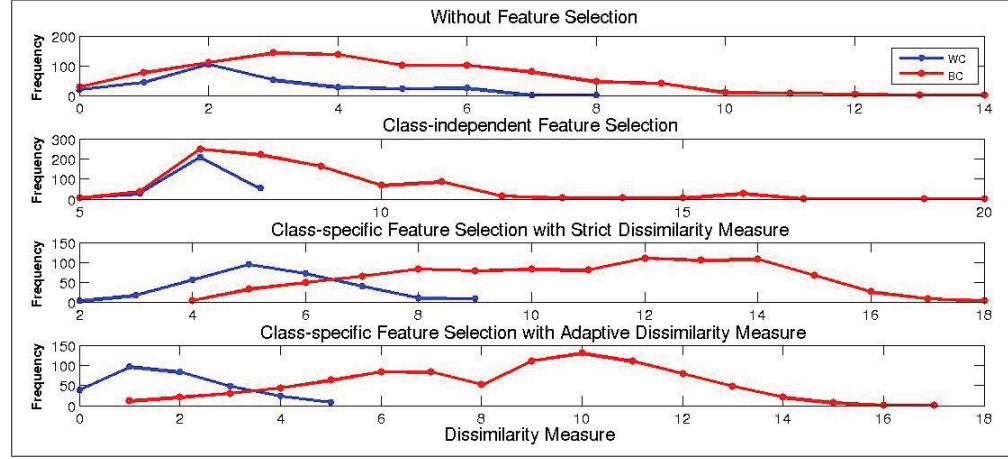


Figure 2.6 Dissimilarity score distribution for a specific user.

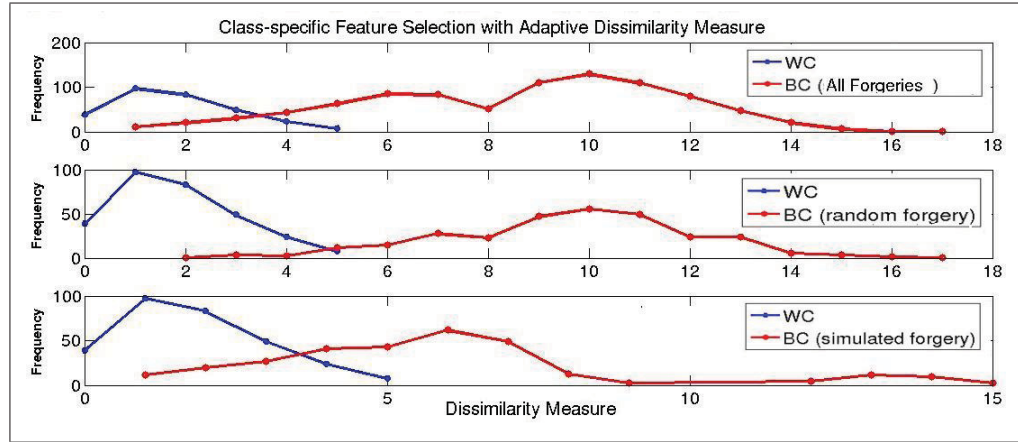


Figure 2.7 Dissimilarity score distribution for different forgery types.

### 2.6.5 Results and discussion

Figure 2.6 illustrates the impact of each processing step on separating the WC and BC clusters, for a specific user. It is obvious that, without feature selection, the distributions are overlapped. Selecting FR based on CID, increases the separation. This validates our hypothesis that DRs

optimized based on high dimensional FRs extracted from large number of classes, relatively, generalize for unseen classes. Running class-specific feature selection, increased the separability. This validates our hypothesis that, the CIRs are adaptable for new classes. Employing the adaptive distance measure, increased the stability of the genuine class. For instance, the maximum dissimilarity score for the genuine class is decreased from 9 to 5. This validates our hypothesis that, the proposed adaptive proximity measure absorb some of the intrinsic signal variability. However, this impact differs for the different forgery types. For instance, in Figure 2.7, it is clear that while the random forgery class distribution is significantly separated, the simulated forgery distribution still has significant class overlap.

Table 2.3 Average Hellinger distance over all Users for the different design scenarios. FS stands for feature selection, and ADM stands for adaptive dissimilarity measure.

| <b>Design Aspect</b>  | <b>Without FS</b> | <b>CI-FS</b> | <b>CS-FS</b> | <b>CS-FS with ADM</b> |
|-----------------------|-------------------|--------------|--------------|-----------------------|
| $\hat{H}_{random}$    | 0.2976            | 0.6093       | 0.6617       | 0.7398                |
| $\hat{H}_{simple}$    | 0.2519            | 0.5531       | 0.6011       | 0.6951                |
| $\hat{H}_{simulated}$ | 0.1466            | 0.4395       | 0.4786       | 0.5907                |
| $\hat{H}_{all}$       | 0.2496            | 0.5590       | 0.5923       | 0.6617                |
| $AUC$                 | 0.6577            | 0.7724       | 0.9328       | 0.9700                |

Table 2.3 shows the average performance, where the average Hellinger distance is computed over the 60 Users, and for the different types of forgeries. It is obvious that each processing step increased the distances between the WC and BC distributions, for all types of forgeries. Average distance of the all forgeries distributions  $\hat{H}_{all}$  is increased from 0.2496 to 0.6617. Also, the average AUC is increased by about 47% (from 0.6577 to 0.9700).

The dissimilarity scores reported above are averaged for all prototypes in the subset  $R_1^{cs}$ . However, class separation differs for the different prototypes (different columns in the dissimilarity matrix). For instance, Figure 2.8 shows distributions of the best and worst prototypes for a specific user. For the worst prototype, a dissimilarity threshold  $\epsilon = 4$  results in  $FRR = 10\%$ ,  $FAR_{random} = 10\%$  and  $FAR_{simulated} = 30\%$ . For the best prototype,  $FRR = 0\%$ ,  $FAR_{random} = 0\%$  and  $FAR_{simulated} = 20\%$ .

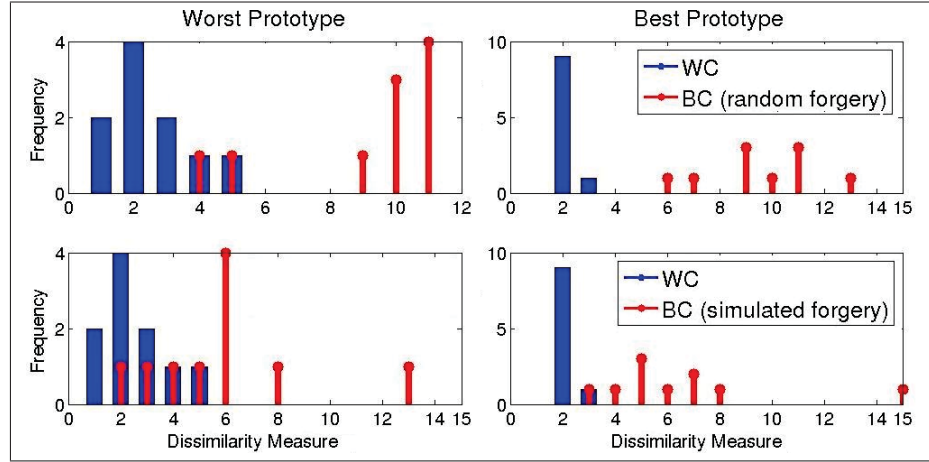


Figure 2.8 Dissimilarity score distributions for different prototypes.

Table 2.4 Impact of the prototype selection on average error rate over all users

| Method            | Rivard<br>(Rivard <i>et al.</i> , 2013) | This work                   |                          |
|-------------------|---|-----------------------------|--------------------------|
|                   |   | Without prototype selection | With prototype selection |
| $FRR$             | 14.36                                   | 5.25                        | 4.83                     |
| $FAR_{random}$    | 0.02                                    | 2.74                        | 0.6                      |
| $FAR_{simple}$    | 0.35                                    | 3.49                        | 1.5                      |
| $FAR_{simulated}$ | 14.24                                   | 33.14                       | 22.33                    |
| $AER$             | 7.24                                    | 11.15                       | 7.32                     |

The recognition rates for the SV (and zero noise error bio-cryptographic) systems are reported in Table 2.4. It is shown that, although the employed representation is concise (only 20 features), and the classifier is so simple (a threshold), the resulting  $AER$  is acceptable. This performance is enhanced through employing the prototype selection step, as the  $AER$  is decreased by about 34% (from 11.15% to 7.32%). Compared to the state-of-the-art results on same SV database, the baseline system by Rivard *et al.*, have shown similar accuracy but with employing 555 features (Rivard *et al.*, 2013). So, applying our proposed DR optimization approach maintained the performance, while decreased the representation complexity by about 96% (from 555 to only 20 features). Moreover digitizing the feature values (as they are represented in 8-bits words for bio-cryptographic encoding), had no impact on the SV accuracy.

Compared to our complete implementation of the FV bio-cryptographic systems (Eskander *et al.*, 2014a), the best reported recognition rate is 8.21%, where 15 FVs are decoded and the majority vote rule is applied to accept/reject the released key. Here, we achieved similar performance with decoding a single FV, but through selecting the best encoding prototype.

## 2.7 Conclusion

This chapter presents an approach for optimizing dissimilarity representations (DRs), that are built using high-dimensional feature representations. The proposed approach produces class-independent DRs generalize well to unknown classes, that are not used for training. In addition, these representations can be further tuned to new classes. This property permits designing global system adaptable to specific classes. Also, the produced DRs are concise, in terms of number of employed feature extractions and prototypes. This allow for designing systems that have limitations on their computational complexity and that rely on high-dimensional FRs, e.g., bio-cryptography systems. In addition, the proposed adaptive user-specific dissimilarity measure and prototype selection methods enhanced the representation efficiency.

The proposed approach is applied to the OLSV problem. It is shown that concise representations produced separable clusters in the dissimilarity space. Accordingly, a simple threshold provides a high level of accuracy with such hard pattern recognition problem, as that of complex systems in the literature. Moreover, the bio-cryptography design problem is formulated as a traditional classifier in the produced DR space, where designing such systems is tractable.

The proposed approach can also be employed as an intermediate tool for designing feature-based classifiers, where the embedded feature representations or the dissimilarity measures feed traditional or distance-based classifiers, respectively. Future work will investigate the power of the proposed approach on other applications (e.g., face recognition, video surveillance, image retrieval, etc). Also, comparing efficiency of the produced DRs to other local distance design methods in the literature is of great interest.

## 2.8 Discussion

A generic approach for learning DR spaces, that are built on top of a traditional FR spaces, is proposed in this chapter. The two-step BFS methodology, employed for dimensionality reduction, provides a way to compromise advantageous of the WD and WI techniques. The first BFS step locates a universal space in which the intra-personal and inter-personal proximities are easily distinguished, so it serves as a WI classification space. Any pair of signatures from new users that are unseen during the design phase can be classified, by computing their proximity in this universal DR space, as being belonging to same writer or different writers. The second BFS step adapts this space to specific users using their training samples. The new user-specific DR space serves as a WD space that is more accurate, as it is tuned to the specific user, and it is sparser as not all of the WI space dimensions are needed to distinguish a specific user from other users. Through this methodology, we answered the first research question: using a limited user-specific training set, how to generate a concise FR space that is stable for the specific user, and its discriminating power generalizes for unseen users? (see the problem statement in Chapter I).

The final DR space is constituted by employing an adaptive dissimilarity measure that measures proximities between the user-specific FRs of a query sample and some selected prototypes. Designing this proximity measure in a DR space provides a way to model and absorb the intrinsic variations of signature images. This answers the second research question: how to design a proximity measure that alleviates the intrinsic variability of the offline signature images?. Projecting the proximity measures to a vectorial DR space, where proximities to different prototypes are the space dimensions, provides a way to select most stable and discriminant prototypes. This answers the third research question: how to select efficient prototypes that produce the DR space?. Since the DR spaces are designed so that the WC and BC proximities constitute compact and isolated clusters, the decision boundary between these clusters is simple. Therefore, simple classifiers might be sufficient for the OLSV task. This property also facilitates design of FV systems, as the error correction codes can be formulated as threshold

classifiers in the DR space. This answers the final research question: how to formulate the OLSV and the FV systems as simple classifiers in the DR space?.

The proposed approach provides a general framework to design OLSV and bio-cryptography systems, while details of each implementation and their performance should be more investigated. In the following two chapters, detailed study of designing OLSV and FV systems, based on the proposed DR learning approach, is presented. Concerning the OLSV problem, although the OLSV systems designed in this chapter have shown acceptable trade-off between accuracy and complexity as compared to complex systems in the literature, they are not secure as signature templates are stored for verification. Writer adaptation of the universal representation is conducted in a reduced DR space, while executing this step in a reduced FR space might provide an alternative secure solution. This might override the curse-of-dimensionality problem, yet the final WD classifier operates in a FR space and therefore no templates are needed for verification. In addition, quality of representations that are adapted in both DR and FR spaces should be compared. The following chapter is focused on these OLSV design issues.



## CHAPTER 3

### A HYBRID WRITER-INDEPENDENT—WRITER-DEPENDENT OFFLINE SIGNATURE VERIFICATION SYSTEM

Standard offline signature verification (OLSV) systems are writer-dependent (WD), where a specific classifier is designed for each individual. It is inconvenient to ask a user to provide enough number of signature samples to design his WD classifier. In practice, few samples are collected and inaccurate classifiers maybe produced. To overcome this, writer-independent (WI) systems are introduced. A global classifier is designed using a development database, prior to enrolling users to the system. For these systems, signature templates are needed for verification, and the template databases can be compromised. Moreover, state-of-the-art WI and WD systems provide enhanced accuracy through information fusion at either feature, score or decision levels, but they increase computational complexity. In this chapter, a hybrid WI-WD system is proposed, as a compromise of the two approaches. When a user is enrolled to the system, a WI classifier is used to verify his queries. During operation, user samples are collected and adapt the WI classifier to his signatures. Once adapted, the resulting WD classifier replaces the WI classifier for this user. Simulations on the Brazilian and the GPDS signature databases indicate that the proposed hybrid system provides comparative accuracy as complex WI and WD systems, while decreases the classification complexity. The content of this chapter was published at the 13th International conference on Frontiers in Handwriting Recognition (Eskander *et al.*, 2012) and the IET-Biometrics Journal, Special issue on Handwriting Biometrics (Eskander *et al.*, 2013c).

#### 3.1 Introduction

Signature verification systems (SV) are employed to authenticate individuals based on their handwritten signatures. There are two modes of operation for SV systems: online and offline. For online systems, users use special devices like special pens and tablets to acquire signature trajectory dynamics such as velocity, pressure, etc. On the other hand, offline SV (OLSV) systems employ digitized signature images for authentication. Only static information can be

acquired from the signature images, producing less informative signals, and hence, a harder pattern recognition task. (Impedovo and Pirlo, 2008),(Batista *et al.*, 2007).

Standard OLSV systems are writer-dependent (WD), where an individual classifier is designed for each user using his enrollment samples (Justino *et al.*, 2001). During verification, only query signature samples are processed by the classifier. Hence, WD systems are secure as no templates are stored for verification. Accuracy of these systems require that users provide enough number of samples to train their classifiers. Hence, the WD approach implies a trade-off between accuracy and user convenience.

A more user-convenient approach is to design a writer-independent (WI) OLSV system. A single global classifier is designed using an independent (development) database prior to enrolling real users to the system. During verification, both query signature samples and at least one signature template are required to produce the classification decision. Hence, users can start using the system with providing a single signature sample. However, such systems are not secure as signature templates are needed for verification. The stored templates can be stolen, deleted or modified. Moreover, these systems do not model the individual signatures, but rather a universal model that should generalize on current and future users. Accordingly, the produced models are complex and ensemble methods are applied for enhanced performance at the expense of significantly increased complexity (Bertolini *et al.*, 2010), (Rivard *et al.*, 2013).

This chapter proposes a solution to compromise between the pros and cons of the WI and WD systems. A hybrid system is proposed where switching between the two approaches is possible. A universal WI classifier is designed with a development database. This enables starting system operation, even if users provide a single signature sample in the enrollment phase. Through operation, signature samples are collected and stored with the user profile. Once enough samples are collected for a specific user, they are used to adapt the universal classifier to this user. From this time on, the resulting WD classifier is used to verify signatures for the specific user. While the universal classifier compares the query samples to the stored user signature templates, the user-specific classifier only uses the query sample to produce the classification decision. Applying this scenario facilitates starting the system without asking the

users to provide high number of enrolling samples. Then, switching to a more secure and less complex operational mode is possible whenever specific number of user samples exist.

To design the WI stage, pairwise dissimilarities are computed between feature representations of intra-personal and inter-personal samples from the development dataset. Then, boosting feature selection (BFS) (Tieu and Viola, 2004) is employed in a dissimilarity representation space (Rivard *et al.*, 2013). To design the WD stage, the resulting global classifier is adapted to each user based on his stored samples. The features embedded in the WI classifier constitutes a universal signature representation that can represent all users. So, subsets of this representation are discriminant for the different users. Accordingly, we tune the universal representation to a user by selecting the subset of feature representation that discriminates him from the others. To this end, stored user samples are represented in the universal feature space. These representations are used to train a WD classifier, by employing another BFS process that produces a more compact and secure classification system.

The proposed system is previously presented in (Eskander *et al.*, 2012). In this chapter, the robustness of the system is further investigated by conducted simulations using the public GPDS signature databases (Vargas *et al.*, 2007), besides the Brazilian database (Freitas *et al.*, 2000). The next section provides an overview of state-of-the-art pure WI and WD OLSV systems. Section 3.3 describes the proposed WI-WD hybrid approach. Section 3.4 describes the experimental methodology applied in this chapter. The experimental results are presented and discussed in Section 3.5.

### **3.2 Pure WD and WI signature verification systems**

The design of WD systems relies on modeling user signatures in a feature representation space. Accuracy of the resulting models is limited by the available samples for training. Enhanced recognition rates of WD systems is recently achieved by training multi-classifier systems (Batista *et al.*, 2010), (Batista *et al.*, 2012). On the other hand, WI systems do not produce models for the individual signatures, but rather a universal model that is valid for all users. In practice, it is impossible to locate a feature representation space in which signatures of all cur-

rent and future users share the same distribution. The dissimilarity concept, where samples that belong to same class are similar, while samples that come from different classes are dissimilar, provides a solution.

The concept of dissimilarity-based classification has been proposed by Elzbieta Pekalska and Robert P.W. Duin., (Pekalska and Duin, 2002). For this approach, the proximity between objects is modeled rather than modeling the objects themselves. Objects belong to a specific class have a shared degree of commonality that could be captured by a dissimilarity value. The dissimilarity measures can be derived in many ways, e.g. from raw (sensor) measurements, histograms, strings or graphs. However, it can also be build on top of a feature representation (Duin *et al.*, 2010).

In the SV context, the WI approach is realized using a dissimilarity (distance) measure, to compare samples (query and reference samples) as belonging to either the same or different user. As most of work on OLSV is feature-based, where many techniques of feature extraction are already proposed (Impedovo and Pirlo, 2008), the employed dissimilarity representations are built on top of a feature representation.

First implementation of the dissimilarity concept to the author identification domain was presented by Cha and Srihari (Cha, 2001). Simultaneous works applied the dissimilarity learning concept on the OLSV problem by Siteargur N. Srihari *et al.*, (Srihari *et al.*, 2004), and Santos and Sabourin *et al.*, (C. Santos *et al.*, 2004). While the first group used the correlation between binary features as a distance measure, the second group employed the Euclidean distance between graphometric feature vectors. In both implementations, the concept of WI-SV system was introduced. Instead of building a single WD classifier for each user using his enrolling signatures, a single global classifier is designed by learning the dissimilarities between signatures of all users. In (C. Santos *et al.*, 2004), a neural network is trained to find the optimal boundary that splits the genuine and forgery classes in the dissimilarity representation space. Later, Oliveira *et al.* (Oliveira *et al.*, 2007), and Bertolini *et al.* (Bertolini *et al.*, 2010) applied the same concept, where they generated different dissimilarity spaces based on different feature representations. A set of SVM classifiers is trained to model the decision boundaries

for the different subspaces. Finally, each SVM is used to produce a partial classification decision, while the final decision relies on the fusion of these partial decisions in the Receiver Operating Curve (ROC) space. Kumar et al. (Kumar *et al.*, 2012) proposed a WI-SV based on surroundedness features.

More recently, Rivard et al. (Rivard *et al.*, 2013) extended the system in (Bertolini *et al.*, 2010) to perform multiple feature extraction and selection. In this work, information fusion is also performed at the feature level. Multiple features are extracted based on multiple size grids. Fusion of these features and projecting them in a dissimilarity space results in dissimilarity representation of high dimensionality. This complex representation is then simplified by applying the boosting feature selection approach (BFS) (Tieu and Viola, 2004). By applying the multi-feature approach with BFS, it was possible to design WI systems with higher performance than the earlier implementations. Moreover, the complex dissimilarity representation (possibly tens of thousands of features) is condensed to a compact universal representation of few hundreds in dimensionality. This representation can classify samples from unknown users, whose signatures had no share in the training process. The accuracy of this WI system could be enhanced through combining multiple decisions based on multiple templates.

Pure WI are insecure due to the need to store reference signatures for verification, however, they are user-convenient as they do not need user samples for training. On the other hand, pure WD are secure but user-inconvenient. Both techniques can produce SV systems with acceptable accuracy, but they are complex due to the fusion of responses from multiple classifiers. The work proposed in this chapter merge the two techniques to overcome the limitations of the pure approaches. To this end, a WI system is designed as in (Rivard *et al.*, 2013), to start system operation with only one enrolling sample. Then, the universal representation embedded in the WI classifier is adapted to each specific user, whenever enough genuine samples are collected. This adaptation step aims to reduce the classification complexity (number of features and number of classifications fused for a decision), and avoiding the need of using reference signatures for verification.

### 3.3 A Hybrid WI-WD signature verification system

#### 3.3.1 Theoretical basis

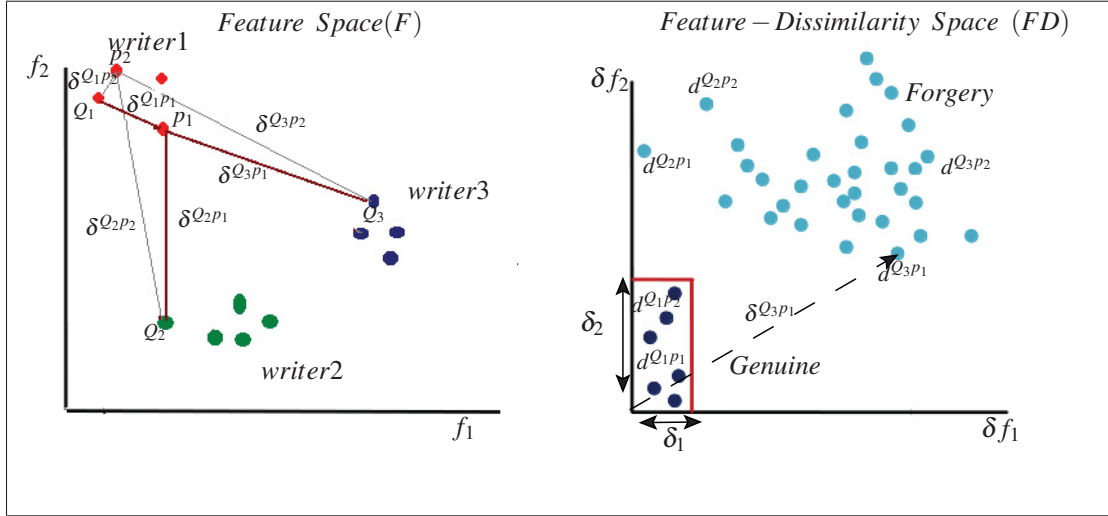


Figure 3.1 Illustration of feature selection in the original feature space (left) and in the feature-dissimilarity space (right).

It is not easy to achieve high generalization accuracy for the OLSV systems, due to the high intra-personal variability and inter-personal similarity of signature images. One approach to tackle this problem, is to select the most stable and discriminant features from a large pool of feature extractions. To illustrate this approach, see Figure 3.1 (left side). In this illustrative example, signatures of three different writers are represented in the feature space. Assume the candidate pool of features is  $F = \{f_n\}_{n=1}^M$ . For simplicity, only two features  $f_1$  and  $f_2$  are shown in this figure, while the dimensionality of this space is  $M$ , which might be a high number for typical representations. For good generalization performance, within-class (intra-personal) distances should be small, while between-class (inter-personal) distances should be large. For instance, assume that writer 1 has two prototypes (templates)  $p_1$  and  $p_2$ . Good generalization implies that, feature representation of any query  $Q_1$  of this writer should be close to his prototypes, while queries of other writers as  $Q_2$  and  $Q_3$  should be far from them.

The proposed approach relies on selecting a condensed feature representation of dimensionality  $N$ , from a very high dimensional space of dimensionality  $M$ , so that distances between intra-personal signature representations are minimized and the inter-personal distances are maximized. Consider the Euclidean distance, so that distance between a signature sample  $Q_j$  and a prototype  $p_r$  is  $\delta^{Q_j p_r}$ :

$$\delta^{Q_j p_r} = \sqrt{\sum_{n=1}^M (\delta f_n^{Q_j p_r})^2} \quad (3.1)$$

where  $\delta f_n^{Q_j p_r} = \|f_n^{Q_j} - f_n^{p_r}\|$ .

Hence, the overall distance between two signatures is an accumulation of the individual distances between every two corresponding features of the signature representations. To increase the separation between the intra-personal and inter-personal distance ranges, we select features that decrease the intra-personal distances and that increase the inter-personal distances. In this illustrative example, it is obvious that features  $f_1$  and  $f_2$  are discriminative. Distances among intra-personal signatures (like  $\delta^{Q_1 p_1}$ ) are generally smaller than the distances among inter-personal signatures (like  $\delta^{Q_2 p_1}$ ). However, in this space it is not clear which feature is more discriminative. With representations of high dimensionality, high number of users, unknown forgeries and a small number of training samples, it is not feasible to select the best features in the feature space  $F$ .

Accordingly, we project this representation on a feature-dissimilarity space  $FD$ , as shown in the right side of Figure 3.1. In this space, distance between each corresponding features, for each pair of signatures, is computed and used as a new set of features  $\{\delta f_n\}_{n=1}^M$ . So, dimensionality of the  $F$  and  $FD$  spaces is the same. A distance  $\delta^{Q_j p_r}$  between a query  $Q_j$  and a prototype  $p_r$  is mapped from  $F$  to  $FD$  as a point  $d^{Q_j p_r}$ :

$$d^{Q_j p_r} = \{\delta f_n^{Q_j p_r}\}_{n=1}^M \quad (3.2)$$

where,  $\delta^{Q_{jPr}}$  is represented by the distance from the origin point to  $d^{Q_{jPr}}$ . Here, the impact of every individual feature on the signature dissimilarities is clear. It is obvious that  $f_2$  is more discriminative than  $f_1$ . For all genuine query samples like  $Q_1$ ,  $\delta f_2^{Q_1Pr} < \delta_2$  and for all forgery query samples like  $Q_2$  and  $Q_3$ ,  $\delta f_2^{Q_{jPr}} > \delta_2$ . On the other hand,  $f_1$  is less discriminant. For the forgery query  $Q_2$ ,  $\delta f_1^{Q_2Pr} < \delta_1$ , same as that for the genuine sample  $Q_1$ . Accordingly, it is easier to rank and select features in the  $FD$  space, as the impact of the individual features on the overall dissimilarity is clear in this space. Moreover, while signatures of users are modeled in the  $F$  space, the proximity between user signatures are modeled in the  $FD$  space. This property maps the multi-class problem, with few training samples per class, to a two-class problem, with more training samples per class. The constituted classes are: the genuine class and the forgery class. Samples of the genuine class result from comparing two signatures of the same person. The forgery class samples result from comparing two signatures of different persons.

Employing the aforementioned feature selection in the  $FD$  space, increases the separation between the genuine and forgery classes, and hence decreases the generalization error. Moreover, Rivard et al., (Rivard *et al.*, 2013) have shown that, classifiers that are designed through feature selection in such dissimilarity representation spaces, can generalize for even users whose signature templates are not used during the design phase. Accordingly, if a classifier is trained with samples from a specific signature database, the same classifier can be used to detect signatures from another database with good accuracy. This observation leads to the concept of WI-SV systems, where an independent (development) database is used to design a classifier, and then signatures of real system users are detected by this global classifier. Good generalization performance could be achieved, when the development database consists in a large enough number of users. As the resulting classifier can classify samples from unknown users, whose signatures had no share in the training process, so the embedded condensed representation of dimensionality  $L < M$  is considered as a population-based representation.

However, such WI-SV systems have some drawbacks. First, in order to model the proximity of samples that belong to a large population, the dimensionality  $L$  of the population-based feature



representation is high, and that produces complex classifiers. Second, the dissimilarity feature representation relies on signature prototypes (templates). Storing user templates in a database might cause security vulnerabilities, as stored signatures can be stolen or edited.

In this chapter, the drawbacks of a WI-SV system are alleviated through adapting it to specific users. The adaptation approach relies on two main hypothesis:

- for each specific user, the population-based representation of dimensionality  $L$  contains a feature subset of dimensionality  $N < L$ , where this more concise representation discriminates the specific user from the other users. The logic behind this hypothesis is that: although the global representation could represent the specific user, not all of the representation dimensions are mandatory for discriminating the user. Also, the importance of representation dimensions differs for the different users. So, re-ordering the features for each user and selecting the most important subset, produces a more compact and maybe more discriminant representation space.
- features that are discriminant when represented in the  $FD$  space are discriminant when represented in the  $F$  space. For instance, if  $\delta f_n$  is discriminant in  $FD$ , then this implies that  $f_n$  is discriminant in  $F$ . This is because that, translation between  $F$  and  $FD$  spaces can be considered as a direct mapping, where this mapping does not impact the Euclidean distances between the signature representations. This property facilitates the design of the user-specific classifier in the feature space, so no signature templates are needed for verification, and hence a more secure WD-SV could be designed. To this end, the population-based representation obtained in the WI-SV design phase, is translated back to a feature space of the same dimensionality  $L$ . Then, user-specific feature selection and classifier design processes are employed in the feature space, to produce a more condensed user-specific space of dimensionality  $N < L < M$ . As the population-based representation is much condensed than the original feature space, so the few training samples available for real system users, can be used to search for the most discriminant space dimensions.

### 3.3.2 System overview

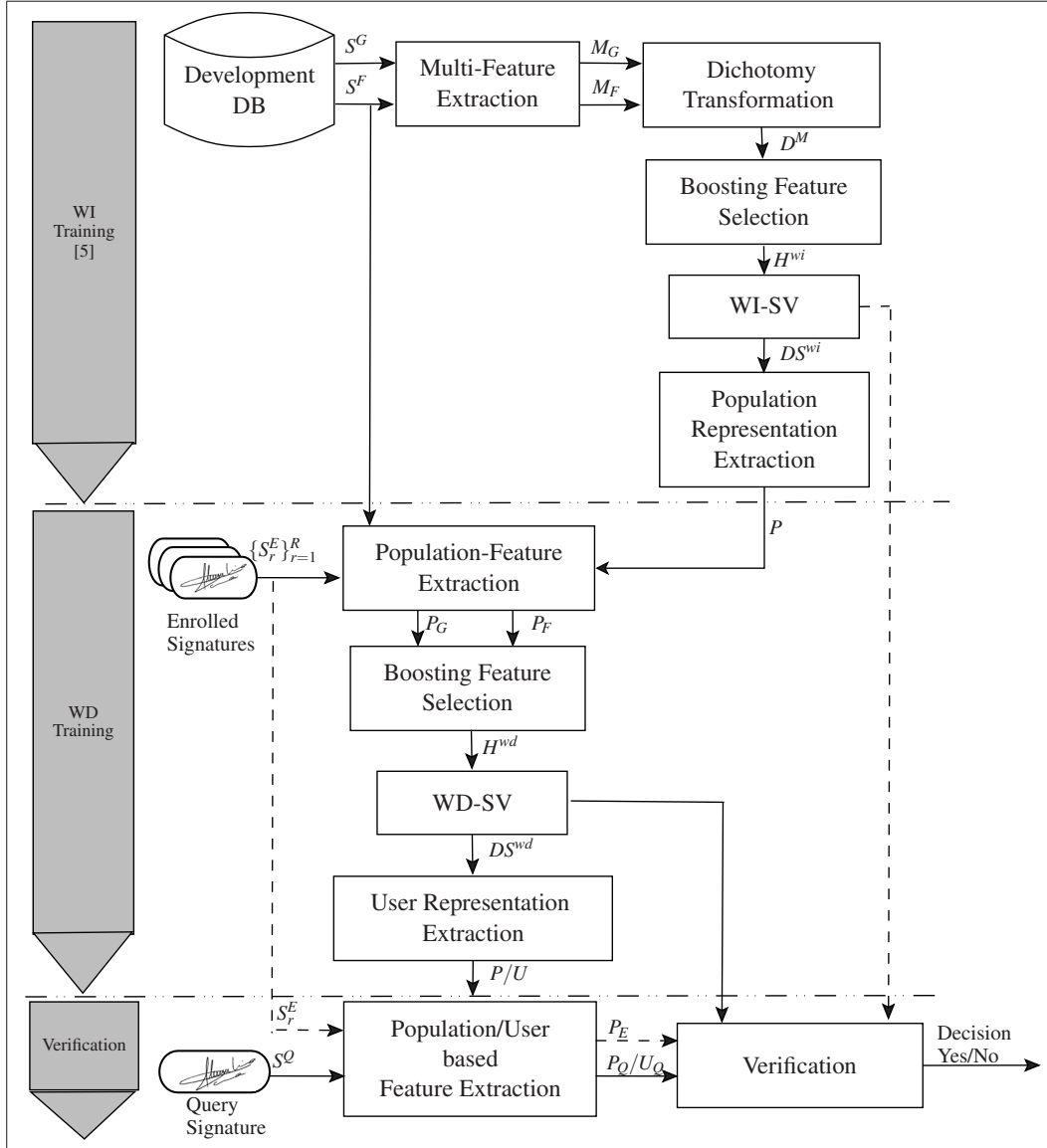


Figure 3.2 Hybrid WI-WD SV system

Figure 3.2 shows a block diagram of the proposed hybrid WI-WD system in training and verification modes. First, a WI-SV sub-system is designed as proposed by Rivard et al. (Rivard et al., 2013). Query samples of recently enrolled users are verified by this sub-system. After collecting a specific number of reference signatures of a user, they are used to adapt this global sub-system to the specific user. To this end, features embedded in the designed WI-SV are

considered as a universal (population-based) representation. The user and forgery samples are translated into this universal space and used to train a WD-SV sub-system. From this time on, signatures of the specific user are verified by his WD-SV sub-system.

### 3.3.3 WI training

A feature level fusion is performed, by employing a multi-type multi-scale feature extractions. In literature, many types of features could be extracted from offline signature images (Impe-dovo and Pirlo, 2008). Any combination of these features may be concatenated into a single high-dimensional representation, and used for the proposed framework. However, we focused on using feature extracted using extended-shadow-code (ESC) (Sabourin and Genest, 1994), and directional probability density Function (DPDF) (Drouhard *et al.*, 1996). Features are extracted based on different grid scales, hence a range of details are detected in the signature image. These features have shown complementary functionality: while ESC detects the distribution of the signature in the spatial space, DPDF detects the orientation of the signature strokes. For more details on the employed the multi-feature extraction technique, see (Rivard *et al.*, 2013).

A development signature database is used to train the WI-SV classifier. To this end, the multi-feature representations  $M_G$  and  $M_F$  are extracted from some genuine signature samples  $S^G$  and forgery signature samples  $S^F$  respectively, where  $M = \{m_n\}_{n=1}^M$ , and  $M$  is the dimensionality of the multi-feature representation. To project these samples into a dissimilarity representation space, dichotomy transformation is applied. For instance, for two samples  $M_i, M_j$ , the dissimilarity feature is:

$$D_{ij}^M = \|M_i - M_j\| = \{\Delta m_n\}_{n=1}^M. \quad (3.3)$$

where  $\Delta m_n = \|m_{i_n} - m_{j_n}\|$ , and  $D^M$  is the dissimilarity representation built on top of the multi-feature representation  $M$ .

It is worth noting that both the multi-feature and dissimilarity representations have the same dimensionality  $M$ . Also, a sample  $D_{ij}^M$  is labeled as a within-class or as a between-class instance, when it results from two genuine signatures of the same user, or from two signatures of two different users, respectively.

To build the WI-SV system, the BFS approach is applied (Tieu and Viola, 2004). This method applies Gentle AdaBoost algorithm (Schapire, 2002) to learn an optimal decision boundary between the within-class and the between-class dissimilarity samples, by boosting Decision Stump ( $DS$ ) weak learners (Iba and Langley, 1992). At a boosting iteration  $n$ , a  $DS$  is designed by locating the best dimension  $d_n$  in the dissimilarity representation space that splits the training samples based on a splitting threshold  $\delta_n$ . The  $DS$  either has positive or negative polarity, depending on the direction of splitting the classes. At a boosting iteration  $n$ , a  $DS_n$  is formulated as:

$$DS_n(d_n) = \begin{cases} p_n^{left} & \text{if } (d_n < \delta_n) \\ p_n^{right} & \text{otherwise} \end{cases} \quad (3.4)$$

$p_n^{left}, p_n^{right}$  represent the confidence of decisions taken by this  $DS$ , when the feature value lies to the left or to the right of the splitting threshold, respectively. Accordingly, each  $DS$  shares in the final classification decision based on its expected accuracy. The boosting process runs for  $T^{wi}$  boosting iterations, and the final decision boundary is defined by:

$$H^{wi} = \sum_{n=1}^{T^{wi}} DS_n^{wi}. \quad (3.5)$$

where  $DS_n^{wi}$  is the  $DS$  designed at boosting iteration  $n$  based on the development data, and  $T^{wi}$  is the number of boosting iteration in the WI training process. Refer to (Rivard *et al.*, 2013) for more detailed algorithms of the WI-SV design process.

This WI-SV sub-system is used to verify user signatures, before switching to the WD mode (the WI verification mode is represented by dotted arrows in Figure 3.2, and illustrated in

Section 3.3.5). However, after collecting enough number of genuine user samples, they are used to adapt the WI-SV sub-system to the user samples. To this end, the WI-SV is used for dimensionality reduction through WI feature selection. The feature representation embedded in a WI classifier is extracted and stored as a population-based representation  $P = \{p_n\}_{n=1}^L$  of dimensionality  $L < M$ , by which signatures of all users are represented. This step reduces the representation dimensionality, and allows for the design of compact user-specific (WD) classifiers.

### 3.3.4 WD training

Although the universal representation  $P$  contains discriminant features for all users, not all dimensions of this space are needed to discriminate specific users from other populations. Moreover, the dissimilarity thresholds selected in the WI system are not optimal for each user. In this design step, selection of discriminant features for each specific user is achieved, while selecting the best splitting threshold in each dimension.

While the WI training phase should be performed in the dissimilarity space (as it is impossible to locate a feature representation space in which signatures of all current and future users share the same distribution), the WD training phase, on the other hand, could be performed in either the dissimilarity or the original feature space. Operating the OLSV system in the feature space is more secure, as no signature references need to be stored for verification. Accordingly, the WD training phase is implemented in the feature space.

To this end, the population-based representation ( $P$ ) of dimensionality  $L$  is used for feature extraction. For each enrolled user,  $R$  signature samples are collected. Both the enrolling samples  $S^E = \{S_r^E\}_{r=1}^R$  and some samples  $S^F$  are selected from the development database (to represent the random forgery class), are represented in the  $P$  feature space as  $P_G$  and  $P_F$  respectively. Finally, a similar BFS process is applied, by using this WD data to model the decision boundary  $H^{wd}$  that splits the genuine and forgery classes, where

$$H^{wd} = \sum_{n=1}^{T^{wd}} DS_n^{wd}. \quad (3.6)$$

where  $DS_n^{wd}$  is the decision stump designed at boosting iteration  $n$  based on the WD training data, and  $T^{wd}$  is the number of boosting iteration in the WD training process.

### 3.3.5 Signature verification

The WI-SV can be used whenever no user samples are available to train a WD classifier. Switching between WI and WD approaches may depend on the availability of sufficient user samples for training.

#### 3.3.5.1 WI-SV mode

This operational mode is illustrated by the dotted arrows in Figure 3.2. A questioned signature  $S^Q$  and a single enrollment sample  $S_r^E$  are represented in population representation space ( $P$ ) of dimensionality  $L < M$  as  $P_Q = \{p_{Q_n}\}_{n=1}^L$  and  $P_E = \{p_{E_n}\}_{n=1}^L$ , respectively. Then, it is classified by the WI-SV system, where

$$WI - SV(D_{QE}^P) = \text{sign}\left(\sum_{n=1}^{T^{wi}} DS_n^{wi}(D_{QE}^P)\right). \quad (3.7)$$

where  $D_{QE}^P$  is the dissimilarity representation of the query sample  $S^Q$  built on top of the population-based feature representation  $P$ :

$$D_{QE}^P = \|P_Q - P_E\| = \{\Delta p_n\}_{n=1}^L. \quad (3.8)$$

where  $\Delta p_n = \|p_{Q_n} - p_{E_n}\|$ .

### 3.3.5.2 WD-SV mode

To authenticate a specific user in this operational mode, the corresponding WD-SV classifier is used. First, the feature representation embedded in the WD-SV is extracted and considered as a user-based representation ( $U$ ) of dimensionality  $N < L < M$ . Then, the query image  $S^Q$  is represented in this concise representation space as  $U_Q = \{u_{Q_n}\}_{n=1}^N$ , and then fed the classifier for recognition, where

$$WD - SV(U_Q) = \text{sign}\left(\sum_{n=1}^{T^{wd}} DS_n^{wd}(U_Q)\right). \quad (3.9)$$

## 3.4 Experimental methodology

Performance of the proposed hybrid WI-WD OLSV system is investigated by considering its two modes of operation:

- **WI-SV mode**—in this mode, the query signature samples are verified by applying Eq.3.7. The objective of investigating this operational mode is to measure the minimum accuracy of the system. In this case, it is assumed that only single signature sample is obtained in the enrollment phase and used for the verification task.
- **WD-SV mode**—in this mode, the query signature samples are verified by applying Eq.3.9. The objective of testing this operational mode is to determine a reasonable number of user samples that produce reliable user-specific classifiers. To this end, performance of the WD classifiers designed with different number of samples is investigated. Suitable switching point between the WI and WD modes is identified by the number of training samples that produce WD classifiers with higher accuracy than the global WI classifier.

### 3.4.1 Signature databases

Two different off-line signature databases are used for proof-of-concept simulations: the Brazilian SV database (Freitas *et al.*, 2000), and the GPDS database (Vargas *et al.*, 2007). While the Brazilian SV database is composed of random, simple and simulated forgeries, the GPDS database is composed of random and simulated forgeries. Random forgeries occur when the query signature presented to the system is mislabeled to another user. Also, forgers produce random forgeries when they do not know neither the signer's name nor the signature morphology. For simple forgeries, the forger knows the writer's name but not the signature morphology. He can only produce a simple forgery using his style of writing. Finally, simulated forgeries imitate the signatures as they have access to a genuine signatures sample.

#### 3.4.1.1 Brazilian database

The Brazilian signatures database contains signatures of 168 users, that were digitized as 8-bit grayscale images over  $400 \times 1000$  pixels, at resolution of 300 dpi. It is split into two parts. The first part contains signatures of the first 60 users. For each user, there are 40 genuine samples, 10 simple, and 10 simulated forgeries. A subset of this part is used for WD training, so it is referenced in this chapter as  $B^{wd}$ . The remaining of this part is used for performance evaluation (see Table 3.3). The second part contains signatures of the last 108 users. For each user, there are only 40 genuine signatures. This part is used for WI training, so it is referenced in this chapter as  $B^{wi}$  (see Table 3.1).

#### 3.4.1.2 GPDS database

The GPDS database contains signatures of 300 users, that were digitized as 8-bit greyscale at resolution of 300 dpi. This database contains images of different sizes (that vary from  $51 \times 82$  pixels to  $402 \times 649$  pixels). All users have 24 genuine signatures and 30 simulated forgeries. It is split into two parts. The first part contains signatures of the first 160 users. A subset of this part is used for the WD training, so it is referenced in this chapter as  $G^{wd}$ . The remaining of this part is used for performance evaluation (see Table 3.4). The second part contains signatures of



the last 140 users. This part is used for the WI training, so it is referenced in this chapter as  $G^{wi}$  (see Table 3.2).

### 3.4.2 Feature extraction

A set of 30 grid scales is used for both of the ESC and DPDF feature types, producing 60 different single scale feature representations. These representations are then fused to produce a multi-feature representation  $M$  of huge dimensionality ( $M = 30, 201$ ) (Rivard *et al.*, 2013).

### 3.4.3 WI training

Table 3.1 The Brazilian Development Database ( $B^{wi}$ ): 108 users x 40 genuine signatures each

| Training set<br>(30 signatures/user)   |   | Validation set<br>(10 signatures/users)  |   |
|--|---|--|---|
| Within-Class                           | Between-Class   | Within-Class   | Between-Class   |
| distances among the 30 signatures/user | distances among 29 signatures/user and 15 signatures of other users | distances among the 10 signatures/user and the 30 signatures of the training set | distances among the 10 signatures/user and 30 signatures selected randomly from other users |
| 108x30x29/2<br>=46,980 samples         | 108x29x15<br>=46,980 samples  | 108x10x30<br>=32,400 samples   | 108x10x30<br>=32,400 samples  |

Table 3.2 The GPDS Development Database ( $G^{wi}$ ): 140 users x 24 genuine signatures each

| Training set<br>(14 signatures/user)   |  | Validation set<br>(10 signatures/users)  |   |
|--|--|--|---|
| Within-Class                           | Between-Class  | Within-Class   | Between-Class   |
| distances among the 14 signatures/user | distances among 13 signatures/user and 7 signatures of other users | distances among the 10 signatures/user and the 14 signatures of the training set | distances among the 10 signatures/user and 14 signatures selected randomly from other users |
| 140x14x13/2<br>=12,740 samples         | 140x13x7<br>=12,740 samples  | 140x10x14<br>=19,600 samples   | 140x10x14<br>=19,600 samples  |

Table 3.1 and Table 3.2 describes the development dataset used in the WI training stage, for the Brazilian and the GPDS databases, respectively. For the Brazilian database, a total of 93,960 samples are used for training, and 64,800 are used as holdout validation set to avoid overfitting. For the GPDS database, a total of 25,480 samples are used for training, and 39,200 are used for holdout validation.

Multi-feature representations of signature images of both the training and validation sets is produced. Then, these representations are fed to the BFS process. The BFS algorithm is set for 1000 max boosting iterations, and 100 early stopping criterion. For the Brazilian database, the constituted WI-SV classifier contained 679 decision stumps, with them only 555 distinct features are used. (i.e.,  $T^{wi} = 679$ ,  $L = 555$ ). For the GPDS database, the WI-SV classifier contained 998 decision stumps, with them only 697 distinct features are used. (i.e.,  $T^{wi} = 998$ ,  $L = 697$ ).

#### 3.4.4 WD training

Table 3.3 The Brazilian WD Database ( $B^{wd}$ ): 60 users x 60 signatures each: 40 genuine+10 simple forgery +10 simulated forgery

| Training set<br>(30 signatures/user) |  | Testing set<br>(30 signatures+ 10 random forgeries/user) |  |
|--------------------------------------|--|--|--|
| Genuine-Class                        | Forgery-Class  | Genuine-Class  | Forgery-Class  |
| Signature subsets of different sizes | Signatures of the training set of the $B^{wi}$ dataset | remaining 10 genuine signatures /user                    | 10 simple+10 simulated +10 random forgery selected randomly from other users in $B^{wd}$ |
| 5,7,9,11,13,15,30 samples            | 108x30<br>=3240 samples                                | 60x10<br>=600 samples                                    | 60x30<br>=1,800 samples  |

Table 3.3 and Table 3.4 describe the data sets used to build the WD classifiers and for performance evaluation, for the Brazilian and the GPDS respectively. To investigate the impact of training samples quantity on the recognition performance (and hence determining a reasonable switching point between the WI and WD modes), different number of samples are used to train

Table 3.4 The GPDS WD Database ( $G^{wd}$ ): 160 users x 54 signatures each: 24 genuine +30 simulated forgery

| Training set<br>(14 signatures/user)      |  | Testing set<br>(40 signatures+ 10 random forgeries/user) |  |
|---|--|--|--|
| Genuine-Class                             | Forgery-Class  | Genuine-Class  | Forgery-Class  |
| Signature subsets of different/user sizes | Signatures of the training set of the $G^{wi}$ dataset | remaining 10 genuine signatures /user                    | 30 simulated +10 random forgery selected randomly from other users in $G^{wd}$ |
| 4,8,12,14 samples                         | 140x14 =1960 samples                                   | 160x10 =1600 samples                                     | 160x40 =6400 samples   |

the WD classifier. The forgery class is represented by genuine signatures from the development database. Genuine and forgery samples are represented in the population ( $P$ ) space and used for training (dimensionality of  $P$  is  $L = 555$  in case of the Brazilian database, and  $L = 697$  in case of the GPDS database). Then, these representations are fed to the BFS process. For the WD training, fixed number of boosting iteration  $T^{wd}$  is used for early stopping. For both databases, we observed saturation in performance around 100 boosting iterations, so the boosting iterations was set to a fixed number (here, the performance is reported for two cases where,  $T^{wd} = 20$ , and 100).

### 3.4.5 Performance measures

The testing sets are illustrated in Tables 3.3 and 3.4. For the Brazilian database, 40 test samples per user are employed. Of them, 10 genuine, 10 random, 10 simple, and 10 simulated forgeries, for a total of 2400 questioned signatures are employed for system evaluation. For the GPDS database, 50 test samples per user are employed. Of them, 10 genuine, 10 random, and 30 simulated forgeries, for a total of 8000 questioned signatures.

The area under ROC curve ( $AUC$ ) and the average error rate ( $AER$ ) are used to evaluate the accuracy of classifiers in this chapter. For  $AUC$  computations, the questioned signatures  $S^Q$  of the test set are processed by a classifier. Its outputs are then sorted, and used as a set of classifier thresholds. Then, the  $GAR$  (genuine accept rate) and  $FAR$  (false accept rate) are

computed for each specific threshold. Finally, the ROC curve is plotted using the generated  $GAR$  and  $FAR$  values, and the  $AUC$  is computed<sup>1</sup>.  $AUC$  classifiers are averaged over all users of the testing dataset. The Average Error Rate ( $AER$ ) is computed as follows:

$$AER = (FRR + FAR_{random} + FAR_{simple} + FAR_{simulated})/4. \quad (3.10)$$

where  $FRR$  is the False Rejection Rate, and  $FAR_{random}$ ,  $FAR_{simple}$ , and  $FAR_{simulated}$  are the False Accept Rates when verifying random, simple, and simulated forgeries, respectively. (for the GPDS only random and simulated forgeries are considered).

Computational complexity of the designed classifiers is evaluated by the total number of feature values ( $TFV$ ) that are extracted and processed to produce the final classification decision (Bunke and Kandel, 2002), where

$$TFV = \sum_{i=1}^n m_i x_i. \quad (3.11)$$

$n$  is the number of partial classification decisions that cooperate to produce the final decision,  $m_i$  is the number of features per sample processed by a classifier  $i$ , and  $x_i$  is the number of signature samples processed by a classifier  $i$ .

### 3.5 Simulation results

Simulations reported in this section address two main objectives:

- feasibility of using the proposed system in its both verification modes: practical switching point between the two modes is identified. Also, robustness and computational complexity of the system are investigated.
- comparing the proposed hybrid system with other pure WI and WD systems in the literature.

---

<sup>1</sup>AUC values are used here only as quality indicators for the constituted classifiers. During the  $WD$  training phase, we do not take any design selections (like early stopping, decision threshold, etc), based on the ROC curves and their  $AUC$  values. Such decisions are only taken based on the development dataset in the  $WI$  design phase.

### 3.5.1 Performance of the WI and WD verification modes

The AUC and AER are computed for both the WI-SV and some WD-SV classifiers designed with different number of training samples and boosting iterations. Based on these measures, we observe a suitable point to switch between the two modes. This point is globally determined for all system users, and identified by the number of training samples that produce a WD classifier outperforms the baseline WI-SV classifier. Robustness of the system is investigated by comparing the verification performance for both the Brazilian and the GPDS experimental databases. Finally, computational complexity and the computer processing times are compared for the two verification modes.

#### 3.5.1.1 Brazilian database

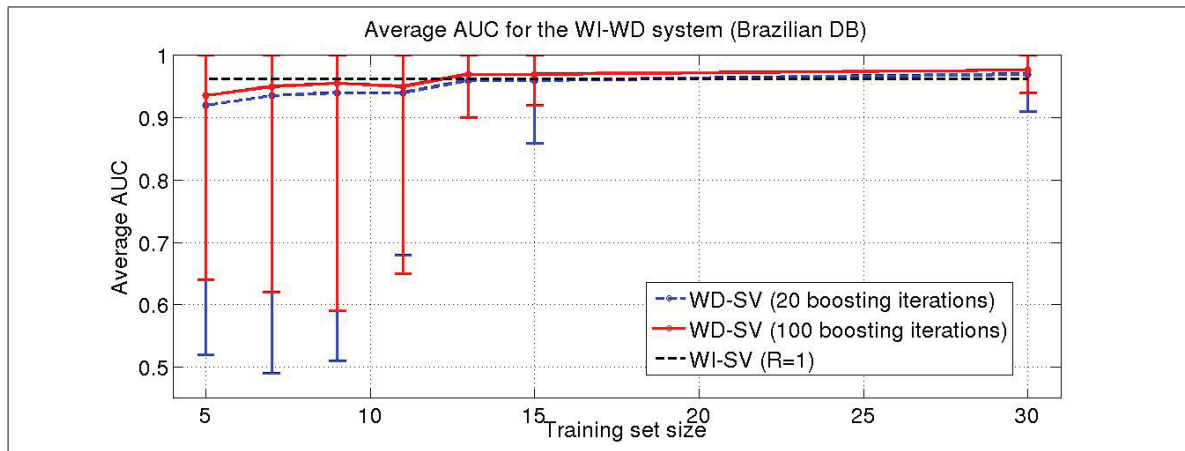


Figure 3.3 Average AUC of ROC curves for the WI and WD classifiers for the Brazilian database. The points represent the average AUC over the 60 users, and the vertical bars represent the range between the maximum and minimum AUCs for the 60 users.

Figure 3.3 shows the  $AUC$  of the WI-SV classifier and some WD-SV classifiers, designed with different number of boosting iterations and with different training set sizes. It is shown that with only 5 training samples, and only 20 boosting iterations, the average  $AUC$  of the WD classifier is 0.923. The classifier performance increases when increasing both training set size and boosting iterations (the average and minimum values of  $AUC$  are increasing). A WD

classifier with 13 training samples has same AUC as the WI classifier<sup>2</sup>. WD classifiers trained with more samples outperforms the WI classifier .

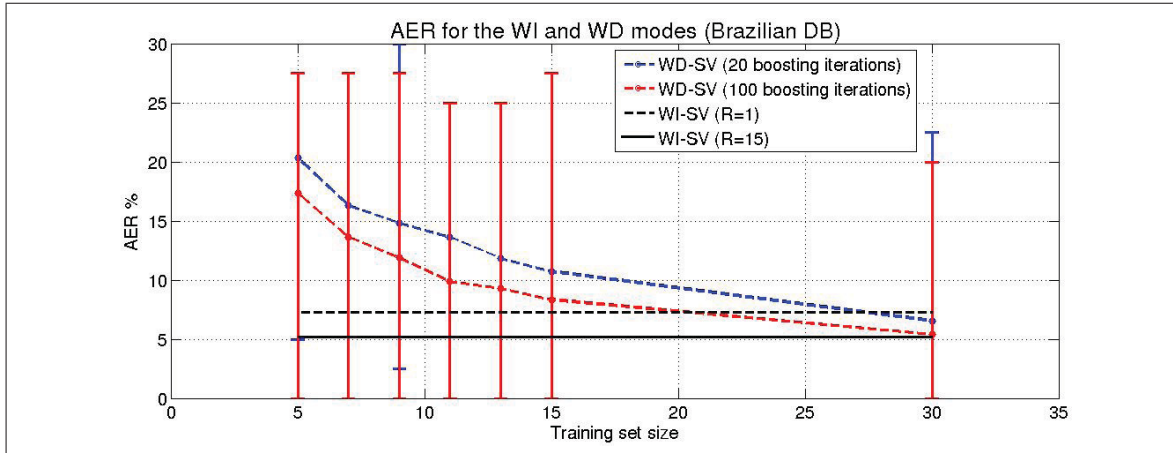


Figure 3.4 AER for the WI and WD classifiers for the Brazilian database. For 100 boosting iterations, the secure WD classifier trained with 20 training samples has same performance (AER=7.24%) as that of insecure WI classifier (tested with a single template ( $R = 1$ )). WD classifiers trained with more samples outperforms the WI classifier.

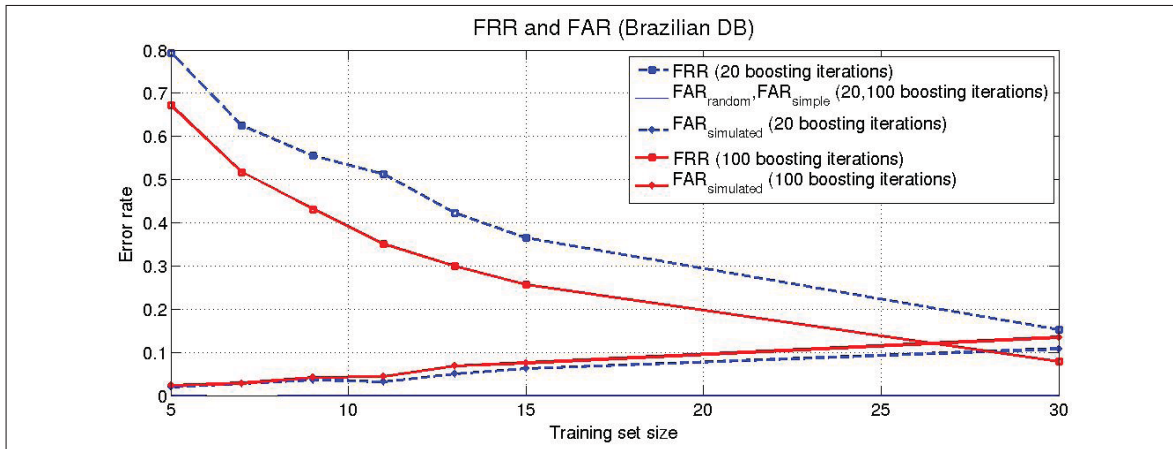


Figure 3.5 FRR and FAR for the WD-SV mode for the Brazilian database.

Figure 3.4 shows the a *AER* for the different classifiers. Same trend is noticed as that of Figure 3.3. However, higher number training samples (20) is needed to produce a WD-SV with smaller

<sup>2</sup>For the WI-SV system, the ROC curve is computed for a single WI classifier tested with a single signature template ( $R=1$ ). Here, no techniques for decision fusion is applied to generate enhanced ROC curves.

$AER$  than that of the baseline WI-SV. The reason of this difference in determining the optimal switching point is because we do not tune the decision thresholds of the classifiers. So, some accurate classifiers may produce high  $AER$  when the decision threshold is not optimized. For 100 boosting iterations, the secure WD classifier trained with 20 training samples has same performance ( $AER=7.24\%$ ) as that of insecure WI classifier (tested with a single template ( $R = 1$ )). Also, a WD classifier trained with 30 training samples has same performance ( $AER=5.38\%$ ) as that of insecure WI classifier (tested with 15 templates ( $R = 15$ )).

Figure 3.5 shows the  $FRR$  and  $FAR$  for different level of forgeries, when WD-SV classification mode is employed. It is clear that, although the  $FRR$  decreases with using more training samples and boosting iterations, the  $FAR_{simulated}$  increases. However, the  $FAR_{simple}$  and  $FAR_{random}$  are neglected when compared to the other error rates.

### 3.5.1.2 GPDS database

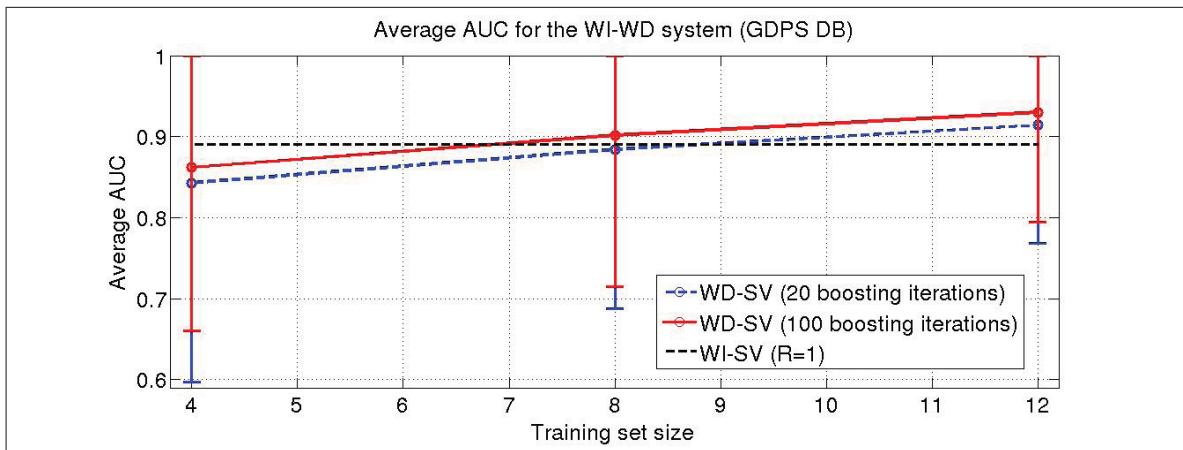


Figure 3.6 Average AUC of ROC curves for the WI and WD classifiers for the GPDS database. Classifiers performance increase when increasing both training set size and boosting iterations. The WD classifier with only 8 training samples has same AUC as the WI classifier. WD classifiers trained with more samples outperforms the WI classifier.

Figure 3.6 shows the a  $AUC$  of the WI-SV classifier and WD-SV classifiers for the GPDS database. Robustness of the proposed system is clear as similar performance trend is shown as that of the Brazilian database. With both databases, classifier performance increases when



increasing both training set size and boosting iterations (the average and minimum values of  $AUC$  are increasing). However, for the GPDS database, the system has shown lower performance.

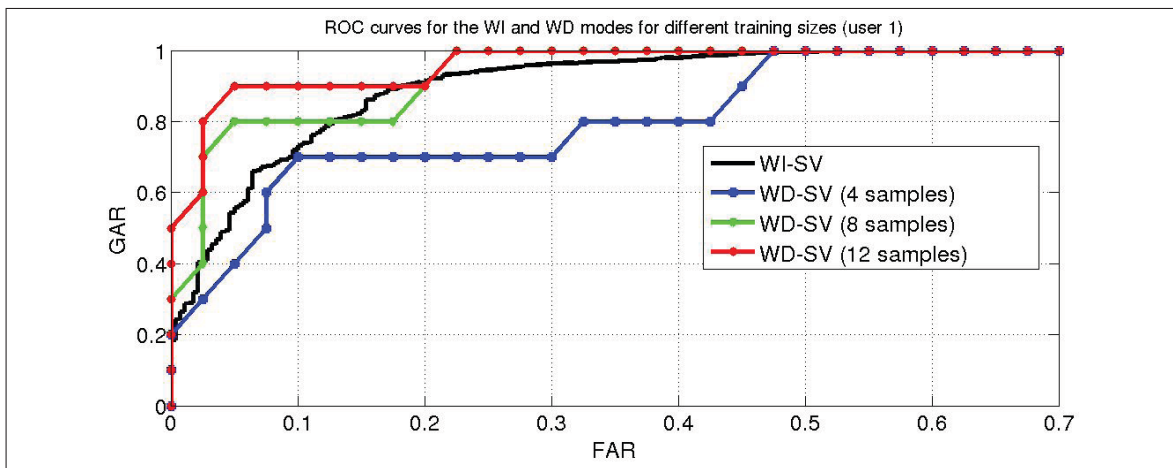


Figure 3.7 ROC curves for the WI and WD modes for different training sizes (user 1).

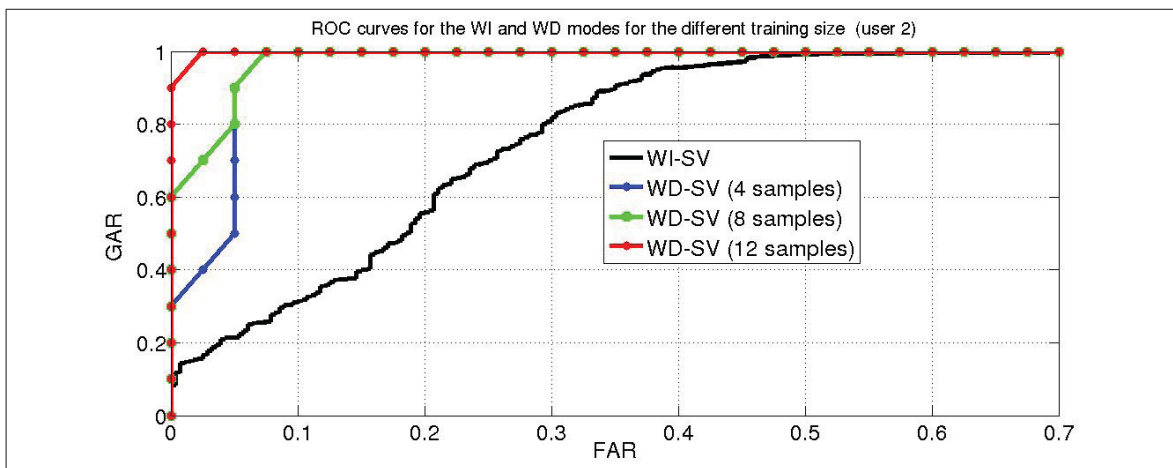


Figure 3.8 ROC curves for the WI and WD modes for different training sizes (user 2).

Although a global training size can be determined, and it results in a better accuracy than that of the original WI-SV system, the best training size differs for the different users. For instance, Figure 3.7 shows the ROC curves for a specific user, where eight training samples produce WD-SV systems with higher performance than that for the WI-SV system. Figure 3.8 shows



the ROC curves for another user where only four training samples could produce WD-SV systems with higher performance than that for the WI-SV system, as the performance of the WI classifier is very weak for this specific user. Future work is needed to investigate possibility of employing user-specific training size.

Moreover, the classification decisions of both the WI and WD classifiers can be fused in the ROC space, and might produce better performance. For instance, a recently fusion method called IBC, proposed by Khreich et al. (Khreich *et al.*, 2010), could be employed to fuse decisions from multiple ROC curves. Such fusion of both modes might be beneficial for the starting operational period (before the switching point to the WD is reached). So, future work will investigate fusing the two classifiers during this operational period.

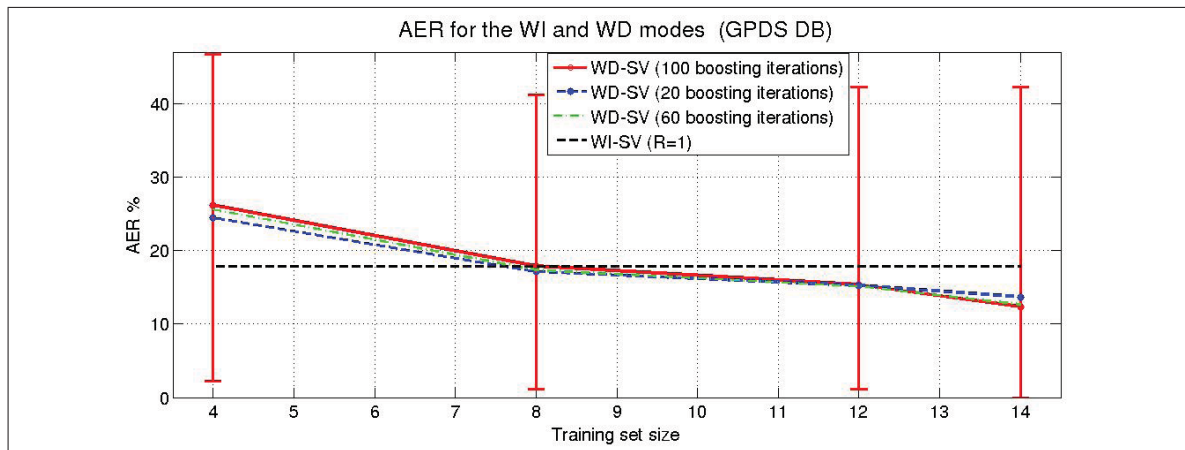


Figure 3.9 AER for WD-SV verification mode, for the GPDS database. The WD classifier with 8 training samples has same AER as the WI classifier (tested with a single template ( $R = 1$ )). WD classifiers trained with more samples outperforms the WI classifier.

Figure 3.9 and Figure 3.10 show the  $AER$  and  $FRR/FAR$  for the different classifiers, respectively. Different than the Brazilian database, signatures of the GPDS seem to be less stable as overfitting occur with only 20 boosting iterations. In Figure 3.10, it is clear that the  $FRR$  increases after 20 boosting iterations, while the  $FAR_{simulated}$  decreases. Also, although the average  $AER$  is acceptable (about 12.5% with 14 training samples and 100 boosting iterations), some users have shown inaccurate performance (maximum AER is about 42%).

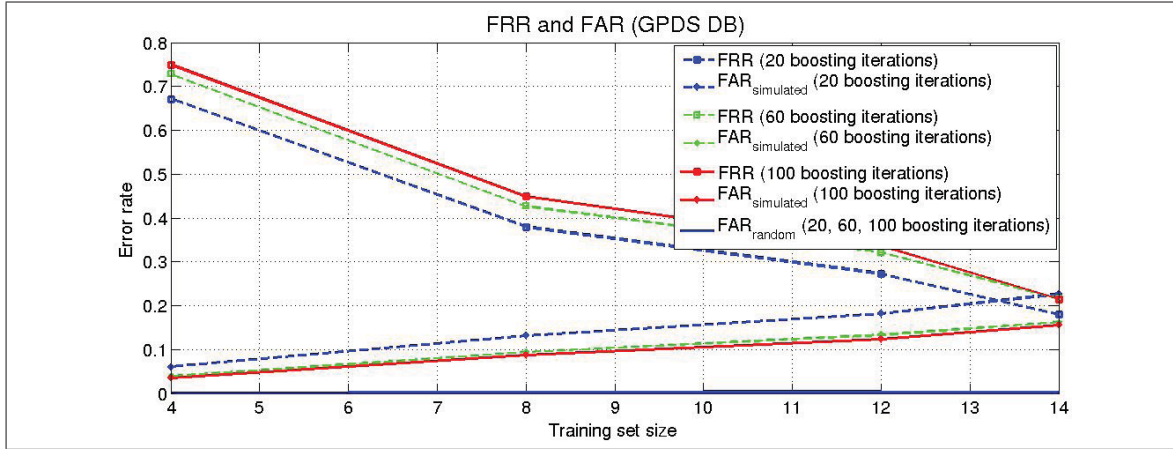


Figure 3.10 FRR and FAR for the WD-SV mode for the GPDS database.

The explanation of the lower performance of the system, when tested on the GPDS database, than that of the Brazilian database is: while the signature images of the Brazilian database have fixed size, the GPDS database includes images with various sizes. Hence, the population-based feature representation ( $P$ ) that is designed based on the development dataset, may not generalize to some users in the WD dataset, whose signature sizes differ significantly. Accordingly, the proposed system is expected to produce high classification accuracy when employed in real-world SV applications, where signature samples have fixed size as they come from a same type of document, i.e., checks from a specific bank.

### 3.5.1.3 Computational complexity

To compare the computational complexity of the two modes of verification, we investigated the complexity of a WI-SV and a WD-SV with similar accuracy. For instance, for the Brazilian database, the WI-SV when tested with fifteen template ( $R = 15$ ) is compared to the WD-SV when trained with 30 samples, as both have similar accuracy. For the WI-SV system, the classifier produces the classification decision based on processing of 555 features extracted from a query sample and a template. Hence, when 15 templates are used (1 query + 15 templates = 16 signature images are processed), the  $TFV$  is 8880 (see Eq.3.11). On the other hand, the WD-SV system frequently produces decisions based on a single classification operation. Only a query sample is used for feature extraction, where about 40 features are processed by the

Table 3.5 Overall error rates provided by systems designed by the Brazilian database

| System                                       | Type    | # templates |              | FRR   | FAR    |        | AER  |
|--|---------|-------------|--------------|-------|--------|--------|------|
|  |         | training    | verification |       | random | simple |      |
| 1.Santos (C. Santos <i>et al.</i> , 2004)    | WI      | -           | 5            | 10.33 | 4.41   | 1.67   | 8.02 |
| 2.Bertolini (Bertolini <i>et al.</i> , 2010) | WI      | -           | 15           | 11.32 | 4.32   | 3.00   | 6.28 |
| 3.Rivard (Rivard <i>et al.</i> , 2013)       | WI      | -           | 15           | 9.77  | 0.02   | 0.32   | 5.19 |
| 4.Justino (Justino <i>et al.</i> , 2001)     | WD      | 30          | -            | 2.17  | 1.23   | 3.17   | 7.87 |
| 5.Batista (Batista <i>et al.</i> , 2010)     | WD      | 30          | -            | 9.83  | 0.00   | 1.00   | 7.79 |
| 6.Batista (Batista <i>et al.</i> , 2012)     | WD      | 20          | -            | 7.50  | 0.33   | 0.50   | 5.46 |
| 7.Proposed                                   | WI mode | -           | 1            | 14.36 | 0.02   | 0.35   | 7.24 |
|  | WD mode | 30          | -            | 7.83  | 0.016  | 0.17   | 5.38 |

classifier. Hence, the  $TFV$  is about 40. Accordingly, adaptation of this WI system to different users reduces the computational complexity by about 99.5%. While similar accuracy is achieved by employing either the WI-SV or the WD-SV systems, the later is lighter and more secure.

We observed consistent computer simulations outcomes. The total verification processing time is dominated by the representation extraction process. While the classifiers compute the classification decision in about  $10^{-5}$  s, the extraction time for population-based representation  $P$  and the user-based representation  $U$ , for a single image, are 0.25 s and 0.02 s, respectively. Hence, the representation extraction time for WI-SV verification mode is about 4 s (for 16 images), and it is about 0.02 s for the WD-SV mode. Accordingly, adaptation of the WI-SV to specific users reduced the verification time by about 99.5%.

### 3.5.2 Comparisons with systems in the literature

The proposed systems is compared with pure WI systems on the Brazilian database (C. Santos *et al.*, 2004),(Bertolini *et al.*, 2010), (Rivard *et al.*, 2013), and on the GPDS database (Kumar *et al.*, 2012). Also it is compared with pure WD systems on the Brazilian database (Justino *et al.*, 2001),(Batista *et al.*, 2010),(Batista *et al.*, 2012) and on the GPDS database (MA *et al.*, 2005),(Vargas *et al.*, 2008),(Solar *et al.*, 2008),(Batista *et al.*, 2012).

#### 3.5.2.1 Brazilian database

Table 3.5 compares the proposed WI-WD system to some pure WI and WD systems that are tested on the Brazilian database. All systems are investigated using the same data set and similar testing protocol, and results are reported in terms of  $AER$ . The first 3 systems are WI systems, while the last 3 are WD systems. The WI systems do not use user signature templates for training, however, an independent (development) signature database is used. It is clear that system 2 outperforms system 1 as it applied information fusion on the decision level, instead of the single classifier in system 1. Also, system 3, that applied information fusion on both the feature and decision levels, outperforms system 2 (both systems applied majority vote

of decisions based on 15 templates). The proposed system, when employed in the WI mode and with using a single template for verification, showed comparable performance of the pure WI-SV systems.

When employed in the more secure WD mode, our system showed similar performance as system 3 (the baseline system of our work), while only single classification decision is executed, instead of fusing 15 classification decisions in the baseline system. Comparing with the WD systems, system 6 has best performance among the other WD systems. Although this system executes a complex dynamic selection of classifiers, the proposed system showed similar accuracy with a single classification operation. For the Brazilian database, the actual accuracy of our system ranges between 5.38% and 7.24%, based on the point of switching between the WI and WD operational modes.

### 3.5.2.2 GPDS database

Table 3.6 compares the proposed system to some pure WI and WD systems that are tested on the GPDS database. Only the first system is a WI system, while the other systems are WD. Although the WI system presented in (Kumar *et al.*, 2012) does not use signature templates of real system users, it used forgery signatures (from the development dataset) in training. For our system, we did not use forgery signatures for training. For systems 2,3, and 4, genuine and forgery samples (of the real system users) are used for training and/or for selecting optimal decision thresholds. This scenario may bias the reported system accuracy, since forgeries are not available during the design of a real-world SV system. System 5 applied similar experimental protocol to ours, where no forgeries are considered available for training. This system showed comparable performance as that of the aforementioned WD system, despite that it did not use forgeries for training. So, system 5 outperforms the earlier systems, however, it applies complex generative-discriminative system with dynamic selection of classifiers. Also, for systems 1,2 and 5 (see second row), the  $ERR$  (Equal Error Rate) is reported. In this case the decision threshold is selected to produce equal values for  $FRR$  and  $FAR_{simulated}$ . Our system showed AER comparable to that of system 5, where the threshold selection is employed based

Table 3.6 Overall error rates (%) provided by systems designed by the GPDS database ( $g$  means genuine and  $f$  means forgery)

| System                                   | Type               | # templates         |              | FRR                     | FAR                        |                         |             | AER                     |
|--|--------------------|---------------------|--------------|-------------------------|----------------------------|-------------------------|-------------|-------------------------|
|  |                    | training            | verification |                         | random                     | simulated               | average     |                         |
| 1.Kumar (Kumar <i>et al.</i> , 2012)     | WI                 | $f$                 | 1            | 13.76                   | –                          | 13.76                   | –           | 13.76                   |
| 2.Ferrer (MA <i>et al.</i> , 2005)       | WD                 | $12g + 12f$         | –            | 14.10                   | –                          | 12.60                   | –           | 13.35                   |
| 3.Vargas (Vargas <i>et al.</i> , 2008)   | WD                 | $12g + 12f$         | –            | 10.01                   | –                          | –                       | 14.66       | 12.33                   |
| 4.Solar (Solar <i>et al.</i> , 2008)     | WD                 | $12g + 12f$         | –            | 16.40                   | –                          | –                       | 14.20       | 15.30                   |
| 5.Batista (Batista <i>et al.</i> , 2012) | WD                 | $12g$<br>$12g$      | –<br>–       | 19.19<br>16.81          | 9.81<br>–                  | 47.25<br>16.88          | –<br>–      | 25.42<br>16.84          |
| 6.Proposed                               | WI-mode<br>WD-mode | –<br>$12g$<br>$14g$ | 1<br>–<br>–  | 26.42<br>27.25<br>18.06 | 0.0056<br>0.0031<br>0.0031 | 27.04<br>18.17<br>22.71 | –<br>–<br>– | 17.82<br>15.24<br>13.96 |

on an independent (development) database. For the GPDS database, the actual accuracy of our system ranges between 13.96% and 17.82%, based on the point of switching between the WI and WD operational modes.

### 3.6 Conclusions and future work

A solution to compromise between pure writer-dependent (WD) and writer-independent (WI) offline signature verification (OLSV) systems is proposed. A universal WI classifier is designed with a development database, to enable starting system operation with few signature templates. Switching to a more secure, less complex, and more accurate WD operational mode is possible whenever enough samples are collected for a specific user. Adaptation of the WI classifiers to specific users is achieved through tuning the universal signature representation to each user, while training his WD classifier.

Simulation results on two real-world offline signature databases confirm the feasibility and robustness of the proposed approach. The initial universal (WI) verification mode showed comparable performance to that of state of the art offline SV systems. The final user-specific WD verification mode showed enhanced accuracy with decreased computational complexity. Only a single compact classifier produced similar level of accuracy (*AER* of about 5.38% and 13.96% for the Brazilian and the GPDS databases, respectively) as complex WI and WD systems in literature. In addition, the produced WD classifiers are more secure than the baseline WI classifiers, eliminating the need to store user templates for verification.

Future work will investigate the ability to enhance the system accuracy by employing other features, and learn from independent forgeries, during the WI training. Also, user adaptation of classifier parameters (like decision thresholds, image size normalization, and training size), will be investigated. Fusion of both WI and WD system modes will be investigated, in order to enhance the performance during the initial operational period of the system. In this chapter, the user samples are assumed to be collected offline. A more practical scenario, where authenticated samples are used to train the WD classifiers online, will be investigated.

### 3.7 Discussion

The proposed DR optimization approach is modified in the chapter, so the final WD classifier operates in a FR space where no templates are needed for verification. The first BFS runs in a high-dimensional DR space using signature samples of an independent development database. The resulting representation has a reduced dimensionality, and it is more discriminant than the original representation. As the resulting representation demonstrated better generalization for the exploitation database, with users are not seen during the design phase, so it is considered universal (population) representation. Designing a WD classifier by limited number of user-specific training set, is more tractable in this reduced space. Different than the OLSV implementation that is reported in Chapter II, here we use the first BFS only for dimensionality reduction, then we return back to a reduced FR space where the final WD classifier designed. It is demonstrated that running the user-specific BFS in the FR space results in more accurate classifiers. Compared to the DR-based OLSV accuracy reported in Chapter II, OLSV systems designed in the reduced universal FR space have shown better accuracy (AER is reduced from 7.32% to 5.38%). In this context, the proposed DR learning approach can be employed as a tool for feature selection, where designing classifiers in the reduced dimensional spaces is more tractable (see appendix I)

The designed OLSV systems might enforce authenticity of transactions, however confidentiality and integrity security aspects needs a bio-cryptographic implementation based on the offline signature images. In the following chapter, the proposed DR learning approach is employed for designing signature-based FV systems, where implementation details and performance analysis are investigated.



## CHAPTER 4

### A BIO-CRYPTOGRAPHIC SYSTEM BASED ON OFFLINE SIGNATURE IMAGES

In bio-cryptography, biometric traits are replacing traditional passwords for secure exchange of cryptographic keys. The Fuzzy Vault (FV) scheme has been successfully employed to design bio-cryptographic systems as it can absorb a wide range of variation in biometric traits. Despite the intensity of research on FV based on physiological traits like fingerprints, iris, face, etc., there is no conclusive research on behavioral traits such as offline handwritten signature images, that have high inter-personal similarity and intra-personal variability. In this chapter, a FV system based on the offline signature images is proposed. A two-step boosting feature selection (BFS) technique is proposed for selecting a compact and discriminant user-specific feature representation from a large number of feature extractions. The first step seeks dimensionality reduction through learning a population-based representation, that discriminates between different users in the population. The second step filters this representation to produce a compact user-based representation that discriminates the specific user from the population. This last representation is used to generate the FV locking/unlocking points. Representation variability is modeled by employing the BFS in a dissimilarity representation space, and it is considered for matching the unlocking and locking points during FV decoding. Proof of concept simulations involving 72000 signature matchings (corresponding to both genuine and forged query signatures from the Brazilian Signature Database) have shown FV recognition accuracy of about 97% and system entropy of about 45-bits. The content of this chapter was published at the IEEE Workshop on Computational Intelligence and Identity Management (Eskander *et al.*, 2011), the International workshop on Emerging Aspects in Handwritten Signature Processing (Eskander *et al.*, 2013b), and Information Sciences (Eskander *et al.*, 2014a).

#### 4.1 Introduction

The concept of bio-cryptography for enhanced security exploits the benefits of biometrics and cryptography in a single construction, while alleviating their vulnerabilities. Biometric systems authenticate users based on their physiological traits like fingerprints, iris, face, etc., or

behavioral traits like voice, gait, handwritings, etc (Jain *et al.*, 2006). Although they guarantee user authenticity, biometric systems are vulnerable to a wide range of attacks such as: vulnerability of biometrics databases, irrevocability of compromised traits and overriding the classifier decision (Uludag, 2006).

On the other hand, cryptographic schemes like encryption and digital signature facilitate confidentiality and integrity of information, but they do not guarantee user authenticity. Known as the key management problem, cryptography keys are vulnerable to theft when secured by weak passwords or when stored as plain tokens (Menezes *et al.*, 1996).

Bio-cryptography has been mainly introduced to alleviate the key management problem in cryptography by using biometric traits to secure the private keys (Uludag *et al.*, 2004). However, it can also be considered as a counter measure against aforementioned attacks on biometric systems. Bio-cryptographic systems provide template protection as no explicit reference traits are used for verification, only secured versions are used. Hence, they facilitate revocability as if the biometric template is compromised, different representation can be extracted, and it generates a new template. Moreover, a bio-cryptographic system can be considered as a biometric classifier with a trusted classifier output. Instead of producing a simple classification label, which can be overridden in classical classifiers, bio-crypto systems produce cryptography keys through a protected mechanism.

There are three main bio-cryptography schemes namely, key-release, key-generation and key-binding (Nandakumar *et al.*, 2007). In key-release systems, both biometric and cryptography keys are stored separately, and the crypto-key is released to genuine users based on classical biometric authentication. This technique does not guarantee template security. In key-generation systems, crypto-keys are generated directly from the biometric traits. It is not easy to generate strong, random, and invariant cryptographic keys from the correlated and unstable biometric traits. In key-binding systems, classical crypto-keys are coupled with biometric keys. They cannot be decoupled without applying a genuine sample of the biometric trait. Reliability and security of key-binding techniques surpasses other cryptography schemes, as they protect the biometric templates and produce typical cryptographic keys. However, they involve

a design challenge to absorb the fuzziness of biometric signals resulting from intra-personal variability (IPV) and inter-personal similarity (IPS). This leads to the false rejection of authorized users and acceptance of unauthorized users, respectively. This chapter will focus on a key-binding scheme known as Fuzzy Vault (FV) that efficiently deals with the fuzzy nature of biometric signals (Juels and Sudan, 2002).<sup>1</sup>

The FV construction locks the cryptography key by means of a specific number of locking features extracted from the biometric template. To unlock the vault (and retrieve the crypto-key), unlocking features are extracted from the biometric query sample. A user can authenticate himself (and retrieve his crypto-key) if the unlocking features extracted from his query sample, overlap sufficiently with his locking features. Due to the fuzzy nature of biometrics, overlap between a genuine sample and its reference template may be insufficient to unlock a FV. Moreover, some impostor samples might show sufficient overlap with the biometric template, and hence unlock the FV to unauthorized individuals.

All reliable FV implementations in literature are based on physiological biometric traits, e.g., fingerprint (Clancy *et al.*, 2003)-(Nandakumar *et al.*, 2007), face (Wang and Plataniotis, 2007)-(Nyang and Lee, 2007), 3D face (Franssen *et al.*, 2008), iris (Lee *et al.*, 2008)-(Meenakshi and Padmavathi, 2010), retina (Meenakshi and Padmavathi, 2010), and palmprint (Kumar and Kumar, 2009). No conclusive research was done using the handwritten signatures. In this chapter, we present a complete implementation of the FV scheme based on the offline handwritten signature images. This implementation may enforce security of documents, e.g., bank checks, by means of the embedded signature images.

Handwritten signatures are behavioral biometrics employed in a wide range of forensic and financial applications. Automation of user verification based on his signatures is realized through signature verification systems (SV). There are two modes of operation for such systems: online and offline. In online systems, signatures are captured while the person signs using devices

---

<sup>1</sup>The FV (Juels and Sudan, 2002) and its antecedent fuzzy commitment scheme (Juels and Wattenberg, 1999) may be viewed as an error-tolerant form of Shamir secret sharing (Shamir, 1979). A query biometric sample shall carry enough number of secret shares in order to retrieve the secret encoded in a FV. To restore some of the corrupted shares in noisy biometric samples, a FV employs an error-correction decoder.

that acquire the dynamic characteristics of the signatures like the pressure, velocity, etc. On the other hand, offline systems capture the signature images from the paper after the signing process. As the physical presence of persons is not mandatory in case of the offline signature verification systems, it can be applied in a broader range of applications than the online systems. However, the static features extracted from the signature images may incorporate a lower level of stability and discrimination to design accurate automatic systems, and to enable fully automation of critical processes as found in financial transactions. For a comprehensive review on the developments on this field see (Impedovo and Pirlo, 2008), and (Batista *et al.*, 2007).

Despite of the intensive research in signature verification, the contributions to bio-cryptosystems based on this variable behavioral biometric are limited and concerned mostly with online systems. In particular, Vielhauer *et al.* (Vielhauer *et al.*, 2002), Hoque *et al.* (Hoque *et al.*, 2008), and Yip *et al.* (Yip *et al.*, 2006) generated biometric keys from the on-line signatures. In contrast with bio-cryptography, cancelable biometrics schemes are proposed to enforce "template protection" while no cryptography keys are secured (Ratha *et al.*, 2001). Examples of this approach in on-line signatures are found in (Freire *et al.*, 2008), and (Maiorana *et al.*, 2010).

Some authors have studied the design of FV systems based on handwritten signatures (Freire-Santos *et al.*, 2006), (Freire-Santos *et al.*, 2007). While it is found that FV systems using online signatures have acceptable performance, the authors observed that features extracted from the offline signature images integrate too much similarity between inter-personal samples (high IPS) and too much variance between the intra-personal samples (high IPV), to design a reliable FV system (Freire-Santos *et al.*, 2007).

In this chapter, the concept of dissimilarity representation (Pekalska and Duin, 2002) is employed to produce signature representations with low IPS and IPV. The dissimilarity approach is mainly introduced to differentiate between classes with modeling the proximity between class objects, instead of modeling the objects themselves. We propose that the dissimilarity approach can be employed to design reliable FV systems, where error correction decoders are used. If the dissimilarity between the locking and the unlocking FV points is less than a specific threshold, the decoder succeeds to unlock the bio-cryptographic key. So, functionality of

these decoders can be considered as two-class simple thresholding classifiers that operate in the dissimilarity space.

In literature, the concept of dissimilarity representation is not directly employed to design bio-cryptographic systems. However, some authors proposed methodologies to absorb the dissimilarities between encoding and decoding biometric signals, so that they are within the error correction capacity of the decoder. For instance, Fingerprint-based fuzzy vaults are designed by using some minutia points extracted in the spatial space to constitute the FV locking features. Dissimilarity between locking and unlocking features is decreased by aligning query and template fingerprints (Nandakumar *et al.*, 2007), and by applying an adaptive bounding box during matching the minutia points (Jain *et al.*, 1997).

In our proposed method, the IPS is alleviated by proposing a dissimilarity-based two-step feature selection technique. Feature representations are extracted from the signature images and they are projected on a dissimilarity-representation space, which we call a "feature-dissimilarity" space, where pairwise feature distances are computed. Then, features are selected in this space to produce concise and discriminant user-specific representations, from a huge number of candidate features. The proposed technique employs the boosting feature selection (BFS) approach (Tieu and Viola, 2004), for fast searching in the high dimensional feature-dissimilarity space. In the first step, a recent method by Rivard *et al.* (Rivard *et al.*, 2013) is employed for dimensionality reduction. This method uses signature samples of a large population to learn a relatively low-dimensional feature representation, from a huge number of candidate features extracted based on multi-feature types and different extraction scales. The feature representation selected at this step is a population-based presentation (PR), as it aims for discriminating between different users in the population. In the second step, the PR is filtered for a user-based representation (UR) that discriminates the specific user from the population. This UR is used to lock/unlock the FV.

The IPV is alleviated by introducing the adaptive matching window (AMW) method. Inspired by the fingerprint alignment approach proposed by Nandakumar *et al.*, (Nandakumar *et al.*, 2007) and the adaptive bounding box approach proposed by Jain *et al.*, (Jain *et al.*, 1997), we

model and store the representation dissimilarities during FV encoding. This information is used for adaptively matching the unlocking and locking points during FV decoding. Modeling of the representation dissimilarity is achieved through employing the BFS in the feature-dissimilarity space.

A preliminary study on some of the ideas that are proposed in this chapter has been appeared in Eskander et al., (Eskander *et al.*, 2011), where only the two-step feature selection method is applied. In this chapter, we further investigate the impact of the different steps of this method on the feature representation quality. Also, we introduce the AMW method. This step significantly increases the genuine accept rate (*GAR*), compared to the system represented in (Eskander *et al.*, 2011). Finally, we present a complete description of the bio-cryptographic system, with investigating the impact of the system parameters on its performance.

Proof-of-concept simulations are performed using the Brazilian signature DB (Freitas *et al.*, 2000). The power of the proposed feature representation technique is tested by analyzing the separation between feature representations of the genuine and forgery classes. The importance of each step of the proposed technique is investigated, through measuring its impact on increasing the separation between the two classes. The FV recognition performance is tested by decoding FVs with genuine and signature samples of different level of forgeries (random, simple and simulated forgeries). The trade-off between the system security and its recognition performance is investigated. Also, the proposed cryptographic system is compared to other classical signature verification systems. Finally, applying the decision fusion concept is tested for enhanced recognition performance of the proposed system.

The rest of this chapter is organized as follows. The next section provides some background on FVs as applied to offline signature images. Section 4.3 describes the proposed method for feature representation. Section 4.4 illustrates the proposed FV system based on the offline signatures, along with analyzing its security. The experimental results and the system performance are presented and discussed in section 4.5.

## 4.2 Fuzzy Vaults with offline signatures

### 4.2.1 Fuzzy Vault

A FV scheme locks a secret message  $K$  by means of a user identifier  $T$ . In case that a FV is used as a bio-cryptography construction, the secret message  $K$  is a cryptographic key and the user identifier  $T$  is a biometric template. A FV scheme consists of two processes: 1) FV encoding, and 2) FV decoding.

#### 4.2.1.1 FV encoding

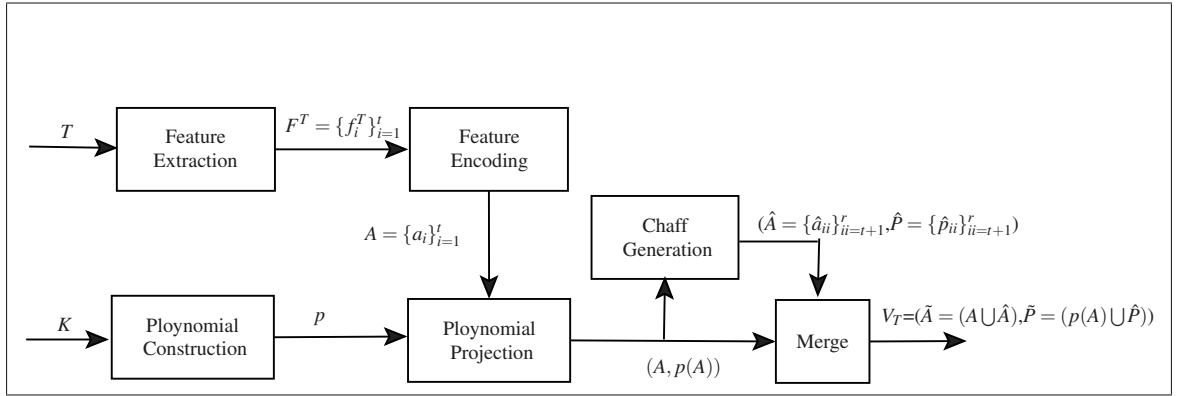


Figure 4.1 Block diagram of the FV encoding process

To encode a FV, features are extracted from a biometric template  $T$  and then they are used to lock the cryptography key  $K$  in a vault  $V_T$ . Figure 4.1 illustrates the processing steps to encode a FV. First, a feature representation  $F^T = \{f_i^T\}_{i=1}^t$  is extracted from  $T$ . The  $t$  elements of  $F^T$  are then quantized in  $l$ -bit strings, to constitute a locking set  $A = \{a_i\}_{i=1}^t$ , where  $a_i$  is the quantized value of the feature  $f_i^T$ .

Second, the set  $A$  locks the secret key  $K$  in a polynomial space as follows:  $K$  encodes a polynomial  $p$  then  $A$  is projected on  $p$ . One way to encode a polynomial  $p$  with a secret  $K$ , is to split  $K$  into equal parts and then use them as polynomial coefficients. For instance,  $K$  is split into  $k + 1$  strings of length  $l$ -bit and constitutes a coefficient vector  $C = \{c_0, c_1, c_2, \dots, c_k\}$ . A

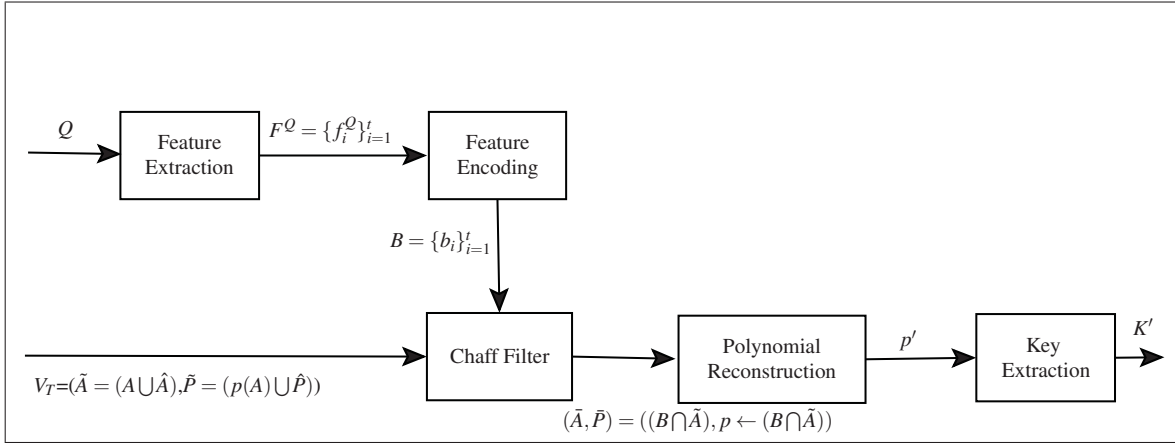


Figure 4.2 Block diagram of the FV decoding process

polynomial  $p$  of degree  $k$  is encoded using  $C$ , where  $p(x) = c_k x^k + c_{k-1} x^{k-1} + \dots + c_1 x + c_0$ . To lock the polynomial  $p$ , and thereby  $K$ , by means of the locking set  $A$ , the polynomial is evaluated for all points in  $A = \{a_i\}_{i=1}^t$  and constitutes the set  $p(A) = \{p(a_i)\}_{i=1}^t$  where  $p(a_i) = c_k a_i^k + c_{k-1} a_i^{k-1} + \dots + c_1 a_i + c_0$ . It is important to note that all computations are done in a finite Galois field  $GF(2^l)$  (Berlekamp and Elwyn, 1968), so that both  $a_i$  and  $p(a_i) \in [0, 2^l]^2$ .

The points  $(A, p(A))$  constitute the genuine vault points. To conceal these points from attackers, chaff (noise) points  $(\hat{A}, \hat{P})$  are generated, where  $\hat{A} = \{\hat{a}_{ii}\}_{ii=t+1}^r$ ,  $\hat{a}_{ii} \in GF(2^l)$ ,  $\hat{a}_{ii} \neq a_i \vee ii \in [t+1, r]$ ,  $i \in [1, t]$ , and  $\hat{P} = \{\hat{p}_{ii}\}_{ii=t+1}^r$ ,  $\hat{p}_{ii} \in GF(2^l)$ ,  $\hat{p}_{ii} \neq p(\hat{a}_{ii}) \vee ii \in [t+1, r]$ . Finally, both the genuine point set  $(A, p(A))$ , and the chaff point set  $(\hat{A}, \hat{P})$  are merged to constitute the vault points  $(\tilde{A}, \tilde{P})$ , where  $\tilde{A} = A \cup \hat{A}$ , and  $\tilde{P} = p(A) \cup \hat{P}$ . The vault  $V_T$  is stored as a user template which consists in the vault points  $(\tilde{A}, \tilde{P})$ , and the vault parameters  $(k, t)$ .

<sup>2</sup>FV decoding relies on error-correction codes. Computations of these codes are based on finite fields that are called Galois fields. Hence, all of the FV items must be represented in a finite field of the same representation size.



#### 4.2.1.2 FV decoding

To learn  $K$  from the vault  $V_T$ , the genuine set  $(A, p(A))$  should be firstly isolated by filtering the chaff points  $(\hat{A}, \hat{P})$  out of the vault set  $(\tilde{A}, \tilde{P})$ . Then any subset of only  $k+1$  genuine points in  $(A, p(A))$ , could be used to reconstruct the polynomial  $p$  of degree  $k$ .<sup>3</sup>

Figure 4.2 illustrates the processing steps to decode a FV. A feature representation  $F^Q = \{f_j^Q\}_{j=1}^t$  is extracted from a biometric query sample  $Q$ . The  $t$  elements of  $F^Q$  are then quantized in  $l$ -bit strings, to constitute an unlocking set  $B = \{b_j\}_{j=1}^t$ , where  $b_j$  is the quantized value of the feature  $f_j^Q$ . Then, the chaff points are filtered by matching items of  $B$  against all items in  $\tilde{A}$ . This process results in a matching set  $(\bar{A}, \bar{P}) = ((B \cap \tilde{A}), p \leftarrow (B \cap \tilde{A}))$ , where  $p \leftarrow (B \cap \tilde{A})$  represents the projection of the matching features on the polynomial space.

In the ideal case, all of the chaff points are filtered out and all of the genuine points are isolated. This case occurs when the query sample  $Q$  typically matches the template  $T$ . In such case, each feature encoded in  $B$  locates the corresponding genuine feature encoded in  $A$ , hence  $(B \cap \tilde{A}) = A$ , where  $\tilde{A} = (A \cup \hat{A})$ . In this case, the matching set  $(\bar{A}, \bar{P}) = (A, p(A))$ . On the other hand, due to the fuzzy nature of biometrics, some elements of  $B$  differ from their corresponding elements in  $A$ , so that  $(\bar{A}, \bar{P}) \neq (A, p(A))$ . More specifically, during the chaff filtering process, two types of errors might occur, namely erasures and noise. For both error types, an element  $b_j$  differs from its corresponding element  $a_i$ . However, for the erasures case,  $b_j$  does not match with any of  $\tilde{A}$ , so it does not add any point to the matching set. For the noise case, an element  $b_j$  might equate a chaff element  $\hat{a}_i$ , so that it adds a noise point  $(\hat{a}, \hat{p})$  to the matching set.

Figure 4.3 illustrates the chaff filtering process. In this illustrative example, a polynomial of degree  $k = 7$  is locked in a FV with a locking set  $A = \{a_i\}_{i=1}^{20}$  of length  $t = 20$ . In the decoding phase, the chaff points are filtered out by an unlocking set  $B = \{b_j\}_{j=1}^{20}$ . Each unlocking element  $b_j$  is matched against all items in the vault. If the query  $Q$  and the template  $T$  are identical, each unlocking element  $b_j$  equates its corresponding locking element  $a_j$ , and

<sup>3</sup>Polynomial reconstruction algorithms like Lagrange interpolation needs only  $k+1$  points on a curve to reconstruct a polynomial of  $k$  degree. For instance, only two points are needed to identify a line.

hence adds the genuine point  $(a_j, p(a_j))$  to the matching set. In this example, there are 14 matching points added to the unlocking list. Of them, 10 genuine points and 4 noise points. While, the other 10 genuine points are not added to the matching set since they did not match with their corresponding unlocking elements.

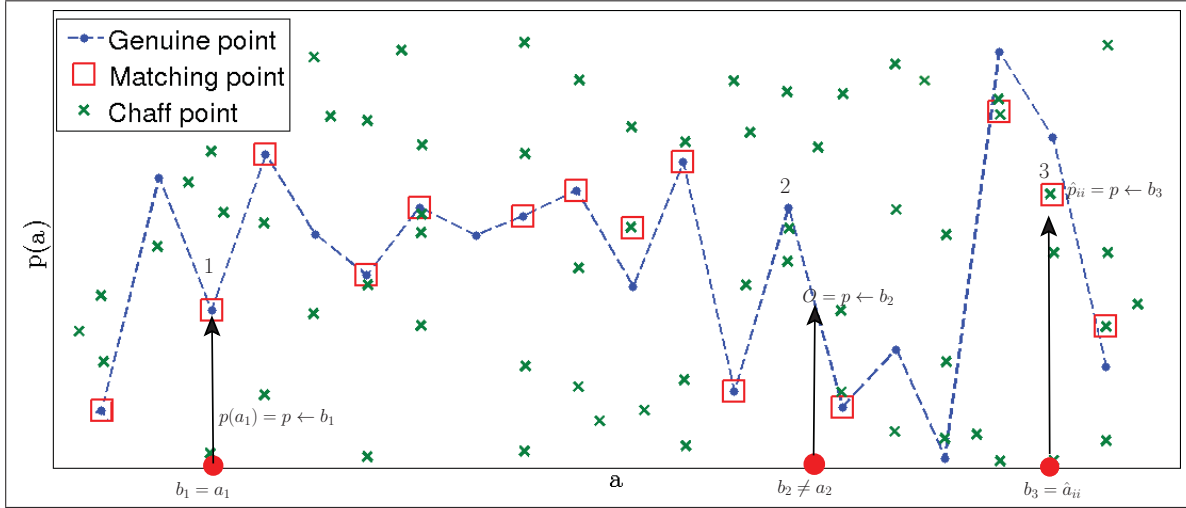


Figure 4.3 Illustration of the chaff filtering Process: 20 genuine points are encoded in the FV, by projecting them on the polynomial space  $p$ . Only 10 of them could be isolated and added to the matching set. For instance, point 1  $(a_1, p(a_1))$  is isolated by means of the unlocking element  $b_1$ , where  $b_1 = a_1$ . While, the other 10 genuine points could not be isolated by means of the corresponding unlocking elements. For instance, point 2  $(a_2, p(a_2))$  could not be isolated from chaffs as  $a_2$  did not match with the corresponding unlocking element  $b_2$ . Also, there are 4 chaff points are added to the unlocking list and considered as noise. For instance, point 3  $(\hat{a}_{ii}, \hat{p}_{ii})$  is incorrectly added to the matching set because  $b_3$  matches with  $\hat{a}_{ii}$

After filtering the chaff points and isolating a matching set  $(\bar{A}, \bar{P})$  of length  $t'$  (in the above example,  $t' = 14$ ), the polynomial reconstruction process uses the matching set to retrieve the encoded polynomial. This process succeeds only if the matching set contains at least  $k + 1$  genuine points. However, even if enough genuine points exists, it is not possible to differentiate the genuine points from the noise points. To overcome this, two approaches could be applied namely, exhaustive search and error correction. In exhaustive search approach,  $N = \binom{t'}{k+1}$  subsets of length  $k + 1$  are constituted from the  $t'$  points of the unlocking list  $(\bar{A}, \bar{P})$ . Each subset is then used to reconstruct a polynomial of degree  $k$ . To locate the right polynomial,

some error detection methods could be applied to check the correctness of the reconstructed polynomial. For instance, CRC codes are computed and compared for both the original and constructed polynomial coefficients (Nandakumar *et al.*, 2007). Also, hash functions could be employed to encrypt both the original and the reconstructed keys, then comparison is done in the encrypted space (Nagar *et al.*, 2011).

For the error correction approach, an error correction decoding like Reed-Solomon (R-S) decoding could be employed (Berlekamp and Elwyn, 1968). The idea is to consider the genuine points  $(A, P)$  as a code word of length  $t$  that encodes a secret message of length  $k + 1$ , where there are  $t - k - 1$  redundancy elements. During the decoding process, some noise is added to this code producing a corrupted version of it. The R-S codes can correct some of these errors<sup>4</sup> and recover the original code, and thereby the encoded secret.

#### 4.2.2 Encoding Fuzzy Vaults with signature images

The aforementioned description of the FV scheme applies for any biometric trait like fingerprint, iris, face, signatures, etc. However, generation of the locking/unlocking sets from the template/query samples depends on the specific biometric traits. Figure 4.4 illustrates a way to generate the locking/unlocking sets from the signature images. In this example, extended shadow codes features (Sabourin and Genest, 1994) are extracted from signature images, and used for generating feature representations  $F^T = \{f_i^T\}_{i=1}^{20}$  and  $F^Q = \{f_i^Q\}_{i=1}^{20}$  from a signature template  $T$  and a signature query sample  $Q$ , respectively. These features are used to encode the locking set  $A = \{a_i\}_{i=1}^{20}$  and the unlocking set  $B = \{b_i\}_{i=1}^{20}$ . In this example, only half of the unlocking elements match with the corresponding elements in the locking set.

We define here a feature encoding space as a two dimensional space consisting of the feature index  $i$  and the feature value  $f_i$ . Matching the template and the query samples is done in this feature encoding space, then the matching set is projected on the polynomial space to filter the chaff points and isolate the genuine FV points. Figures 4.5, 4.6 illustrate the steps to

---

<sup>4</sup>For Berlekamp-Massey algorithm (Berlekamp and Elwyn, 1968), codes of  $t$  elements with  $t - k - 1$  redundancy elements can recover up to  $(t - k - 1)/2$  errors. So, the minimum number of correct elements needed to recover a message of length  $k + 1$  is  $(t + k + 1)/2$ .

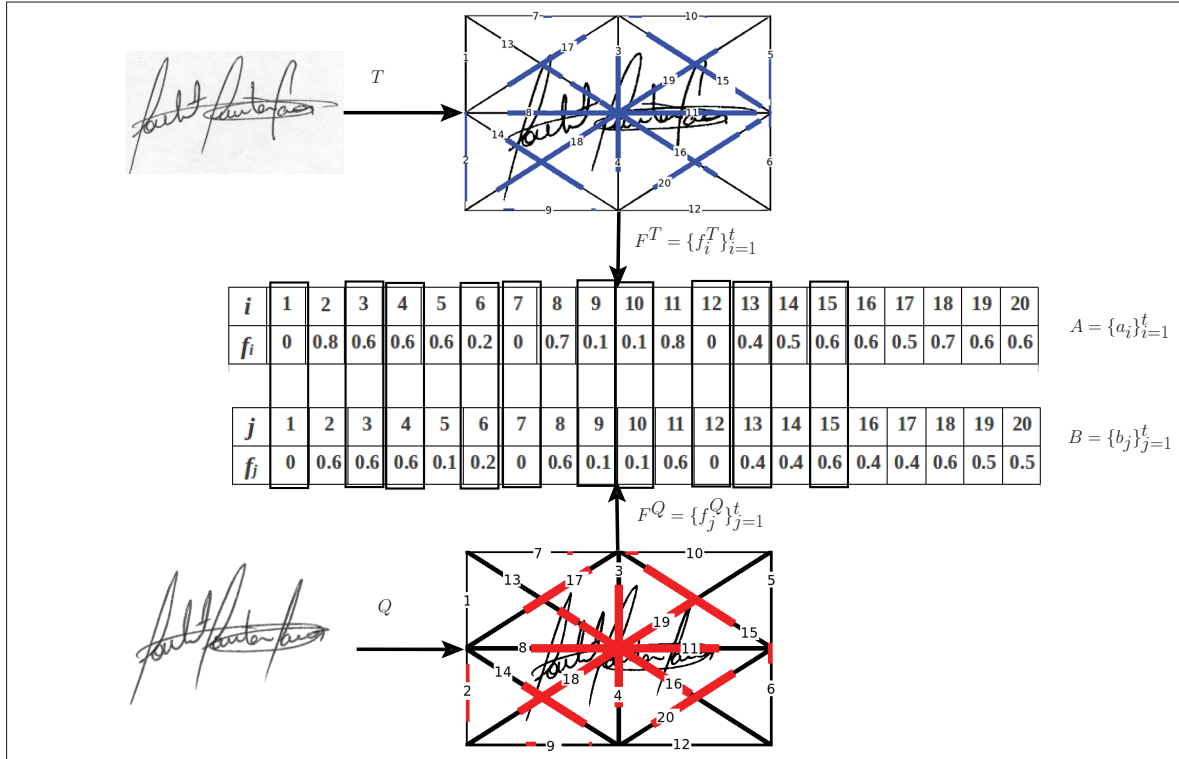


Figure 4.4 Illustration of FV encoding with signature images: the extended shadow code (ESC) (Sabourin and Genest, 1994) features consist in the superposition of bar mask array over the binary image of a handwritten signature. Each bar is assumed to be a light detector related to a spatially constrained area of the 2D signal. A shadow projection is defined as the simultaneous projection of each black pixel into its closest horizontal, vertical and diagonal bars. A projected shadow turns on a set of bits distributed uniformly along the bars. After all the pixels of a signature are projected, the number of on bits in each bar is counted and normalized to the range of  $[0,1]$  to constitute the ESC feature value. The ESC feature vectors  $F^T = \{f_i^T\}_{i=1}^t$  and  $F^Q = \{f_j^Q\}_{j=1}^t$  are extracted from the template signature  $T$  and the query sample  $Q$ , respectively. The FV locking set  $A$  and the unlocking set  $B$  are represented in two-tuples, where  $A = \{a_i\}_{i=1}^t = \{(i, f_i^T)\}_{i=1}^t$  and  $B = \{b_j\}_{j=1}^t = \{(j, f_j^Q)\}_{j=1}^t$ . A locking element  $a_i$  matches an unlocking element  $b_j$  only if they have identical indexes and feature values, i.e., when  $i = j$  and  $f_i^T = f_j^Q$ .

perform this chaff filtering process. First, each unlocking element  $b_j$  is matched in the feature encoding space against all the elements of  $\{a_i\}_{i=1}^t$  (as shown in Figure 4.5). Features encode the locking/unlocking FV points must be stable, so they have equal values when extracted from the template and other genuine query samples. Also, the features must be discriminant, so when extracted from query samples of other users or forgeries, they have different values

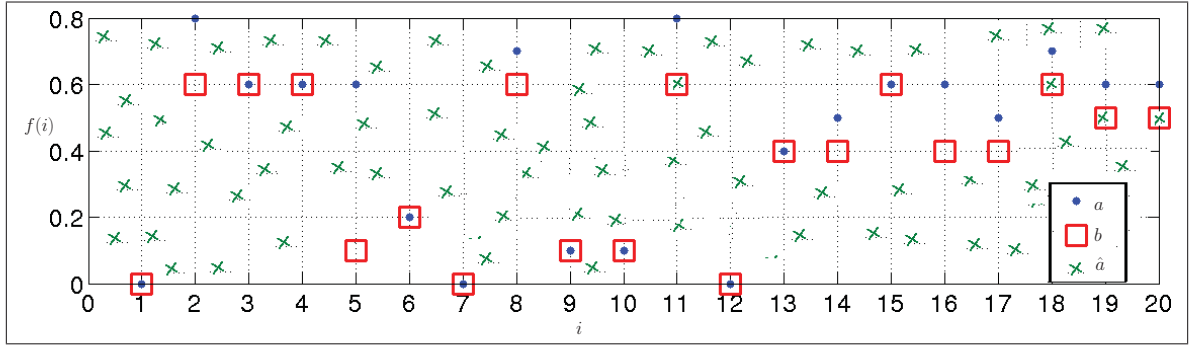


Figure 4.5 Chaff filtering in the feature encoding space: every unlocking point  $b_j$  is matched with all of the vault locking points  $\{a_i\}_{i=1}^t$ , and this step produces the matching points  $\bar{A} = (\{a_i\}_{i=1}^t) \cup \{\hat{a}_{ii}\}_{ii=t+1}^r \cap \{b_j\}_{j=1}^t$ . An unlocking point  $b_j$  succeeds to isolate its corresponding locking point  $a_j$  only if their feature values are equal, i.e.,  $f_j^T = f_j^Q$ . Also, it might happen that the unlocking point  $b$  equates a chaff point  $\hat{a}$  and adds it as noise to the matching points.

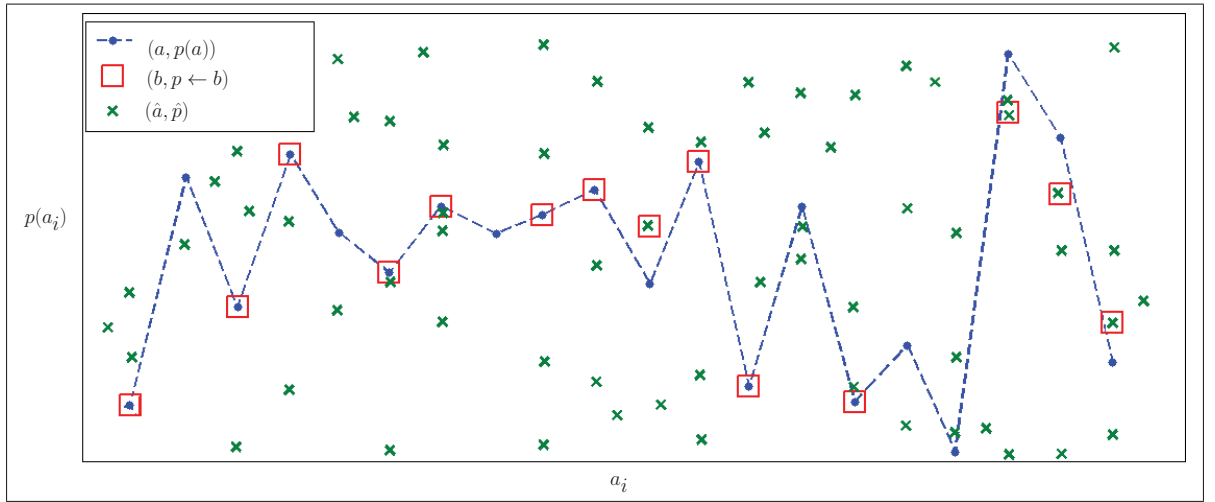


Figure 4.6 Chaff filtering in the polynomial space: only the locking points that are isolated from the chaffs are projected on the polynomial space  $p$ , to isolate the matching set  $(\bar{A}, \bar{P})$ . Due to IPV, some of genuine FV points might not be included in this set. Also, due to IPS, impostor samples might isolate some genuine FV points. Moreover, due to the similarity between some of FV unlocking and chaff points, some chaff points may be incorrectly added to the matching set.

than that extracted from the template. It might happen that an unlocking element  $b$  matches a chaff element  $\hat{a}$ . To avoid such incorrect matches, it is required to generate the chaff points with enough separation from the locking points. Only the matching elements are projected on the polynomial space, and isolates the corresponding genuine points (as shown in Figure 4.6).

It is clear that, a successful FV decoding relies mainly on the amount of overlap between the locking and unlocking sets. Accordingly, to design a reliable FV system, the feature representation of the signature samples should be selected properly, so that two signature representations of the same user have sufficient similarities, while two signature representations of different users should sufficiently differ. In order to consolidate the design aspects of signature representations that are suitable for the FV scheme, we propose the following definitions:

- Similarity score ( $SS$ ): is the number of FV locking points that equate their corresponding points in the unlocking set. As two points  $a_i$  and  $b_j$  are equal only if both of their indexes and feature values are equal, i.e.,  $i = j$  and  $f_i^T = f_j^Q$ . So, the  $SS^Q$  between a template sample  $T$  and the query sample  $Q$ , can be defined as: the number of features that have the same values when extracted from the template  $T$  and the query  $Q$ . Hence:

$$SS^Q = \sum_{i=1}^t (S_i), \text{ where } S_i = \begin{cases} 1 & \text{if } (f_i^Q = f_i^T) \\ 0 & \text{if } (f_i^Q \neq f_i^T) \end{cases} \quad (4.1)$$

- Decoding threshold ( $D\Theta$ ): is the minimum number of genuine FV points found in the matching set  $(\bar{A}, \bar{P})$ , that are needed to decode the FV, and to retrieve the cryptography key  $K$ . This threshold depends on the algorithm used to decode the FV, and on the FV parameters  $(k, t)$ . For instance, for Berlekamp-Massey algorithm (Berlekamp and Elwyn, 1968), a FV with  $t$  locking points that encode a polynomial of  $k$  degree, has a decoding threshold:

$$D\Theta = (t + k + 1)/2 \quad (4.2)$$

Based on the above definitions, a signature representation that encodes/decodes a FV system should satisfy the following condition:

$$SS^Q \begin{cases} \geq D\Theta & \text{if } Q \text{ is a genuine sample} \\ < D\Theta & \text{if } Q \text{ is a forgery sample} \end{cases} \quad (4.3)$$

The above condition implies that the similarity score distributions for the genuine and forgery classes should be separated around the  $D\Theta$ . Accordingly, functionality of the FV decoder is formulated as a two-class simple thresholding classifier that operates in a dissimilarity space. For physiological biometrics, like fingerprints, it is easy to achieve this condition. For instance, the minutia representation of a specific finger is naturally fixed, and representations of two fingers differ sufficiently. In such biometrics, the acquisition process may produce distorted representations of the trait. Some preprocessing steps could be employed to recover the original representation. For instance, fingerprints could be aligned, using helper data, prior to feature extraction (Nandakumar *et al.*, 2007). On the other hand, signatures are behavioral biometrics that have intrinsic variations (high IPV). Also, signatures of different persons may have similarities (high IPS), and more critical, they can be easily imitated by forgeries. It is not easy to identify a representation space with well separated similarity score distributions. We tackle this challenging design issue by proposing a feature selection technique that relies on the dissimilarity representation (Cha, 2001).

### 4.3 Selection of a user-specific signature representation

#### 4.3.1 Feature selection in the feature dissimilarity space

To illustrate the idea behind our proposed feature selection approach, see Figure 4.7. In this example, three signature images are represented:  $T$  is the template signature,  $Q_1$  is a genuine query sample and  $Q_2$  is a forgery query sample. In the left side, signatures are represented in the FV feature encoding space. For simplicity, only two features ( $f_1$  and  $f_2$ ) are shown, while the full representation consists of  $t$  dimensions. On the right side, signatures are represented in the feature dissimilarity space. In this space, a feature is replaced by its distance from a reference value. For instance,  $f_1$  and  $f_2$  are replaced by their dissimilarity representations  $\delta f_1, \delta f_2$ , where  $\delta f_1 = |f_1^Q - f_1^T|$ , and  $\delta f_2 = |f_2^Q - f_2^T|$ . Accordingly, while a point in the feature encoding space represents a signature image, a point in the feature dissimilarity

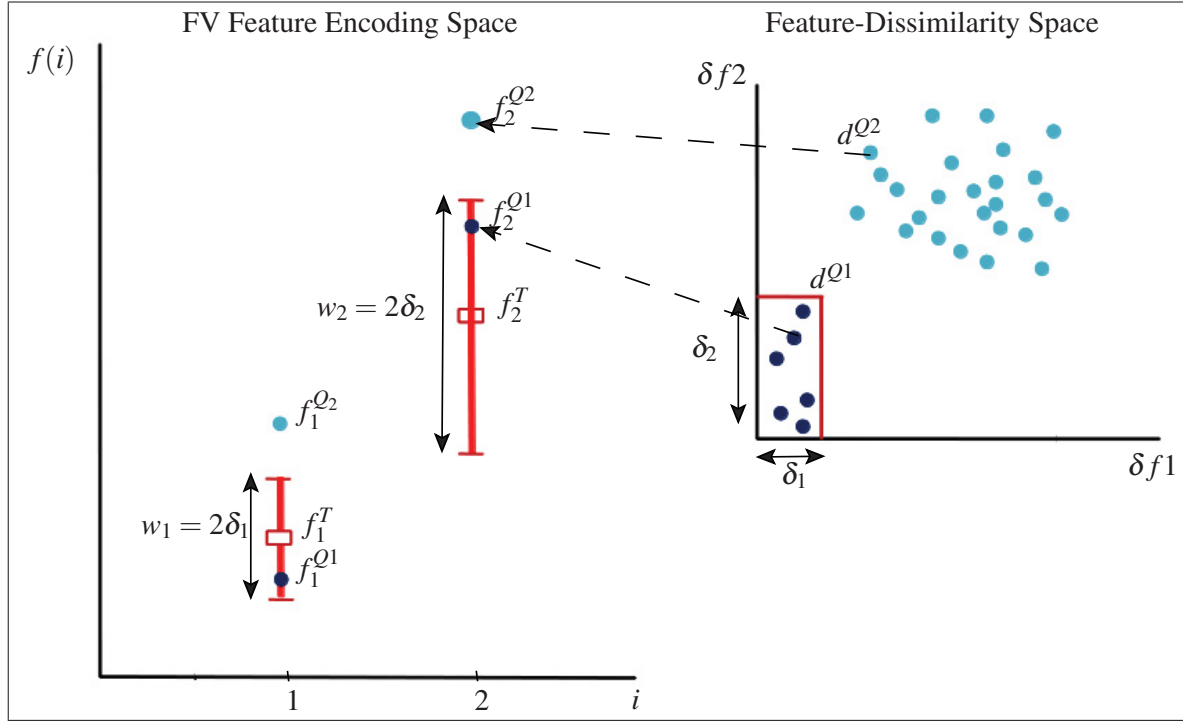


Figure 4.7 Illustration of feature selection in the feature dissimilarity space and in the feature encoding space. In this example,  $\delta f_1$  and  $\delta f_2$  are discriminant features. For instance, for all genuine query samples like  $Q_1$ ,  $\delta f_i^{Q_1} < \delta_i$  and for all forgery query samples like  $Q_2$ ,  $\delta f_i^{Q_2} > \delta_i$ . Unfolding these discriminant dissimilarity features to the original feature encoding space produces discriminant features in the encoding feature space, where the distance between two feature instances is used to determine their similarity. For instance, a genuine feature (like  $f_i^{Q_1}$ ) lies close to the template feature  $f_i^T$ , so they are similar, where closeness here implies that both features reside in a matching window  $w_i = 2\delta_i$ . Features extracted from a forgery image (like  $f_i^{Q_2}$ ) do not resemble the template feature  $f_i^T$ , as they reside outside the matching window  $w_i$ .

space represents the dissimilarity between two different signature images. The point  $d^{Q_1}$  represents the dissimilarity between the genuine signature  $Q_1$  and the template  $T$ , and a point  $d^{Q_2}$  represents the dissimilarity between the forgery signature  $Q_2$  and the template  $T$ , where  $d^{Q_1} = (\delta f_1^{Q_1}, \delta f_2^{Q_1}, \dots, \delta f_t^{Q_1})$ , and  $d^{Q_2} = (\delta f_1^{Q_2}, \delta f_2^{Q_2}, \dots, \delta f_t^{Q_2})$ .

According to the definition of similarity score stated in Eq.4.1, both features  $f_1$  and  $f_2$  are not discriminant as  $SS^{Q_1} = SS^{Q_2} = 0$ . However, the dissimilarity representation of these features is discriminant as the thresholds  $\delta_1$  and  $\delta_2$  perfectly split the genuine and forgery classes. For



instance, for the genuine query samples like  $Q_1$ ,  $\delta f_i^{Q_1} < \delta_i$ . Also, for the forgery query samples like  $Q_2$ ,  $\delta f_i^{Q_2} > \delta_i$ . Based on this idea, the similarity score in Eq.4.1 can be reformulated as:

$$SS^Q = \sum_{i=1}^t (S_i), \quad \text{where } S_i = \begin{cases} 1 & \text{if } (\delta f_i^Q < \delta_i) \\ 0 & \text{otherwise} \end{cases} \quad (4.4)$$

The selected features are used to encode the FV locking/unlocking sets. In the decoding phase, if the unlocking element is close enough to its corresponding locking element, they are considered matching and they isolate the corresponding FV genuine point. For instance,  $f_1^{Q_1}$  is close to  $f_1^T$  so they are matching. On the other hand,  $f_1^{Q_2}$  is not close to  $f_1^T$  so they are not matching. Closeness between two instances of a feature  $i$  is determined by its modeled variability threshold  $\delta_i$ .

#### 4.3.2 A two-step BFS technique with dissimilarity representation

The aforementioned concept could be realized by applying different feature selection algorithms, however, we focus here on employing the boosting feature selection (BFS) approach proposed by Viola et al. (Tieu and Viola, 2004). This approach employs decision-stumps ( $DS$ ) (Iba and Langley, 1992) as weak learners.<sup>5</sup> At each learning iteration, a  $DS$  locates the best representation dimension that splits the two classes around a splitting threshold. If the BFS runs in the feature dissimilarity space, a  $DS_i$  at a learning iteration  $i$ , locates the best dissimilarity representation dimension  $\delta f_i$  that splits the two classes around a splitting variability threshold  $\delta_i$ , so that:

$$DS_i = \begin{cases} 1 & \text{if } (\delta f_i < \delta_i) \\ 0 & \text{otherwise} \end{cases} \quad (4.5)$$

---

<sup>5</sup>A decision-stump ( $DS$ ) is a single-split single-level classification tree. Training of a  $DS$  is equivalent to selection of a single feature that discriminate between two classes based on a splitting threshold.

It is obvious that the functionality of a  $DS$ , when employed in the feature dissimilarity space, simulates the similarity measure defined in Eq. 4.4. This explains why we employed decision stumps as weak learners in the proposed BFS approach.

The learning (boosting) process continues by computing the classification error and weights the samples based on the current error. The misclassified samples get higher weights for the next boosting iteration, giving chance to find a dimension that discriminates the hard samples. After  $t$  iterations, a set  $DS$  is constructed where  $DS = \{DS_i\}_{i=1}^t$ . The learning process is successful if the constructed set is discriminant (i.e., if it results in well separated class representations like that are shown Fig 4.7). In this case, the condition for producing a discriminant FV (stated in Eq. 4.3) can be achieved, where the similarity score  $SS^Q$  is computed according to Eq.4.4.

This feature selection task is challenging as we need to select very few number of features to encode the FV locking/unlocking sets ( $t$  should be small, e.g., 20 features). Our proposed solution for that is to enlarge the search space by generating a huge number of features from the signature images. The signatures are represented in a very high dimensional space and that gives a room to find a small number of dimensions that can split the classes and satisfies Eq.4.3. However, learning in such high dimensional space needs a large number of training samples from both the genuine and forgery classes. In practice, the available positive samples collected when a user is enrolled are few. Also, it is not practical to have forged samples for real system users. To overcome the limitations on the available learning samples, we propose a two-step BFS technique. The first step is population-based feature selection (P-FS) that seeks dimensionality reduction. The second step, is user-based feature selection (U-FS) that seeks to select a user-specific feature representation  $F = \{f_i\}_{i=1}^t$ , along with learning its variability  $\Delta = \{\delta_i\}_{i=1}^t$ . The variability vector is used to match FV points during the decoding phase (as will be described in section 4.4).

Figure 4.8 shows a block diagram of the proposed approach for feature representation selection.

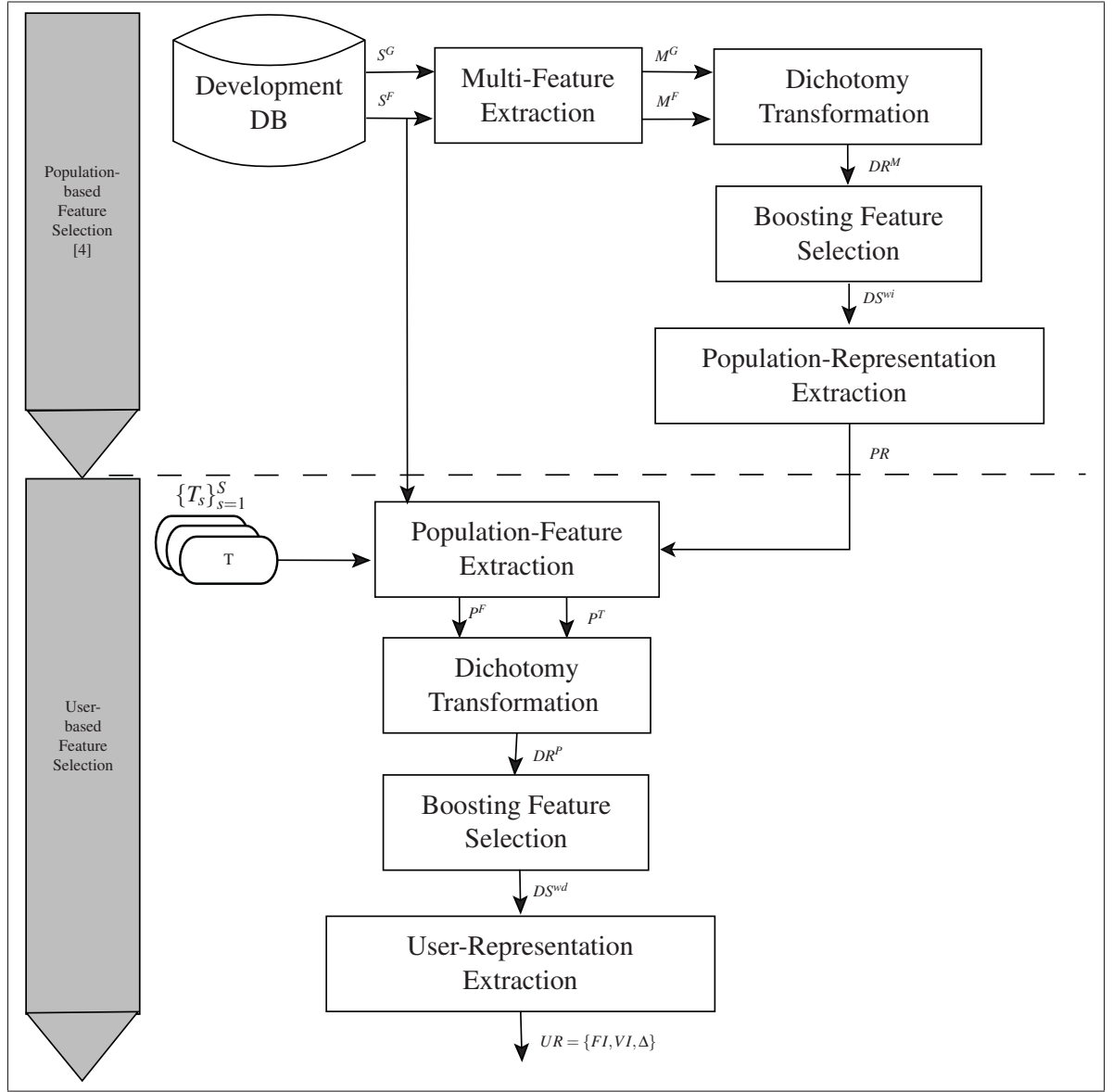


Figure 4.8 Overall block diagram of the proposed approach including population and user-based feature selection

#### 4.3.2.1 Population-based feature selection

This step is developed based on a recent approach for designing of writer-independent signature verification systems (WI-SV) proposed by Rivard et al. (Rivard *et al.*, 2013). In WI-SV, a single signature verification classifier is used to authenticate all system users. As these systems are capable to discriminate between genuine and forgery signatures of any user, even who did not

share in the learning process, so they consist in features that represent the whole population. Accordingly, we extract the features embedded in these classifiers and use them as universal “population-based” feature representations ( $PR$ ).

Figure 4.8 shows the details of this step. A development signature DB, with signatures of none real system users, is used to learn the population-based representation ( $PR$ ). To this end, multi-feature representations  $M^G$  and  $M^F$  of high dimensionality are extracted from some genuine signature samples  $S^G$  and forgery signature samples  $S^F$  respectively, where  $M = (m_1, m_2, \dots, m_M)$  and  $M$  is the dimensionality of the multi-feature representation. To project these samples into the feature dissimilarity space, dichotomy transformation (Cha, 2001) is applied. For instance, for two samples  $M_u, M_v$  the dissimilarity feature is

$$DR_{uv} = |M_u - M_v| = (\Delta m_1, \Delta m_2, \dots, \Delta m_M), \text{ where } \Delta(m_d) = |m_{ud} - m_{vd}|. \quad (4.6)$$

It is worth noting that both the multi-feature and dissimilarity representations have the same dimensionality  $M$ . Also, a sample  $DR_{uv}$  is labeled as a genuine or a forgery instance, when it results from two genuine signatures of the same user, or from two signatures of two different users, respectively.

To learn the  $PR$  representation, Gentle AdaBoost algorithm (Schapire, 2002) is employed. At a boosting iteration  $i$ , a  $DS_i^{wi}$  is designed by locating the best dimension  $\delta f_i$  in the dissimilarity space that splits the training samples based on a splitting threshold  $\delta_i$ , as mentioned in Eq. 4.5. This process runs for  $T^{wi}$  boosting iterations and produces a set of writer-independent decision stumps  $DS^{wi} = \{DS_i^{wi}\}_{i=1}^{T^{wi}}$  that splits the genuine and forgery classes. The distinct features embedded in  $DS^{wi}$  are extracted and stored as a population-based representation (PR) of dimensionality  $L < M$ , by which signatures of all users are represented. This methodology could design a reliable  $PR$  of relatively low dimensionality (few hundred features), that are se-

lected from a feature representation of huge dimensionality (few thousand of features) (Rivard *et al.*, 2013).

#### 4.3.2.2 User-based feature selection

Recently, Eskander et al. (Eskander *et al.*, 2012) extended the work in (Rivard *et al.*, 2013) to design writer-dependent signature verification (WD-SV) systems. In that work, an additional user-specific feature selection step is employed in order to filter a population-based feature representation into a low dimensionality user-specific representation (few tens features). The approach proposed in this chapter extends this work to learn a user-specific representation  $UR$  satisfies Eq.4.3, where the similarity score  $SS^Q$  is computed according to Eq.4.4. The produced  $UR$  is used to encode the FV locking/unlocking sets.

Although the universal  $PR$  contains discriminant features for all users, not all dimensions of this space are needed to discriminate specific users from other populations. Moreover, the dissimilarity thresholds selected in the writer-independent set of stumps  $DS^{wi}$  are not optimal for the different users. In this design step, selection of discriminant features for each specific user is achieved, while selecting the best splitting threshold in each dimension. The PR of dimensionality  $L$  is used for feature extraction. For each enrolled user, sample signatures are collected. Both the enrolling samples  $S^E$  and some samples  $S^F$  are selected from the development DB (to represent the random forgery class), and are represented in the PR feature space as  $P^G$  and  $P^F$  respectively. Then dichotomy transform is applied to transform the features into the feature dissimilarity space. The produced dissimilarity representation  $DR^P$  is then used for feature selection. The same BFS process runs for  $t$  boosting iterations and learns a set of writer-dependent decision stumps  $DS^{wd} = \{DS_i^{wd}\}_{i=1}^t$ , that splits the genuine and forgery classes. The feature representation embedded in  $DS^{wd}$  are stored as a user-based representation (UR) of dimensionality  $t < L < M$ .

As will be described in the next section, the indexes of selected feature  $FI = \{fI_i\}_{i=1}^t$  are used to extract feature representation  $F = \{f_i\}_{i=1}^t$ . Then, a FV point  $a_i$  is constituted by represent  $fI_i$  and  $f_i$  in binary strings of  $l/2$ -bits length, and then both parts are concatenated in a  $l$ -bits string. Accordingly, the indexes vector  $FI$  can not be encoded directly as a part of the FV locking/unlocking sets, as its range may be of few thousands and needs large number of bits when encoded (could not fit in  $l/2$ -bits representation). To overcome this, the feature indexes vector  $FI = \{fI_i\}_{i=1}^t$  is mapped to another vector of virtual indexes  $VI = \{vI_i\}_{i=1}^t$ , where a  $vI_i$  can be represented in a  $l/2$ -bit string. Finally, the vectors  $FI, VI$ , and  $\Delta = \{\delta_i\}_{i=1}^t$  constitutes the user representation matrix  $UR$ , that takes part in both of the proposed FV encoding and decoding processes (as will be described in the next section).

#### 4.4 A Fuzzy Vault system for offline signatures

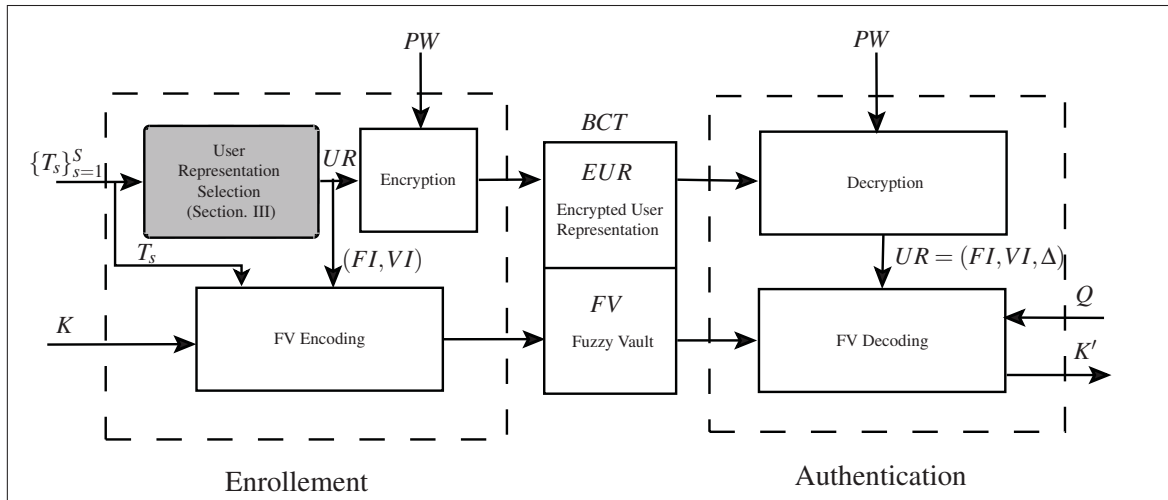


Figure 4.9 Block diagram of the proposed FV system for offline signature images.

The proposed system consists of two main sub-systems: enrollment and authentication (see Figure 4.9). In the enrollment phase, some signature templates  $\{T_s\}_{s=1}^S$  are collected from the enrolling user. These templates are used for the user representation selection, as described in Section 4.3. The user representation selection process results in a user representations matrix

$UR = (FI, VI, \Delta)$ , where  $FI = \{fI_i\}_{i=1}^t$  is the vector of indexes of the selected features,  $VI = \{vI_i\}_{i=1}^t$  is a vector of indexes mapping represented in  $l/2$ -bits, and  $\Delta = \{\delta_i\}_{i=1}^t$  is the vector of expected variabilities associated with the selected features. This matrix is user specific and contains important information needed for the authentication phase. Accordingly,  $UR$  is encrypted by means of a user password <sup>6</sup>  $PW$  and then stored as a part of user Bio-Cryptography Template ( $BCT$ ). Then, the user parameters  $FI$  and  $VI$  are used to lock the user cryptography key  $K$  by means of a single signature template  $T_s$  in a fuzzy vault  $FV$ .

In the authentication phase, the user password  $PW$  is used to decrypt the matrix  $UR$ . Then, the vectors  $FI$ ,  $VI$  and  $\Delta$  are used to decode the FV by means of the user query signature sample  $Q$ . Finally, the user cryptographic key  $K$  is finally restored.

#### 4.4.1 Enrollment process

The enrollment sub-system uses the user templates  $\{T_s\}_{s=1}^S$ , the password  $PW$ , and the cryptography key  $K$  to generate a bio-cryptography template (BCT) that consists of the fuzzy vault  $FV$  and the encrypted user representation matrix  $EUR$ . The user representation selection module generates the  $UR$  matrix as described in the Section 4.3.

The FV Encoding module (illustrated in Figure 4.10) describes the following processing steps:

- a. the virtual indexes  $VI = \{vI_i\}_{i=1}^t$  are quantized in  $l/2$ -bits and produces a vector  $X^T = \{x_i^T\}_{i=1}^t$ .

---

<sup>6</sup>In literature, a password is used to harden the FV system by encrypting the biometric features and the FV (K.Nandakumar *et al.*, 2007). While, in the proposed system, we use the password in a different way: it encrypts the user representation model  $UR$  that is stored with the FV. In the authentication phase, a user has to apply the correct password to restore his feature representation model. Then, he must apply a genuine signature sample from which features are extracted based on  $UR$ , and they are used to produce the FV unlocking set.

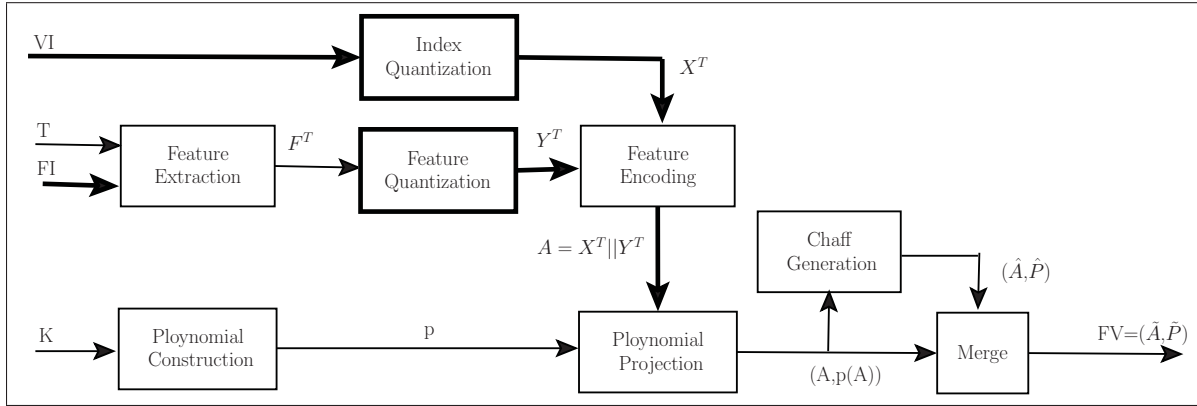


Figure 4.10 Block diagram of the proposed FV encoding process. The bold lines highlight the modules where specific modifications apply to the standard encoding process illustrated in Figure 2.1. A pre-selected user-specific feature model  $FI$  is used for feature extraction and it is mapped to virtual indexes  $VI$ . FV locking points are generated by concatenating the feature values to their indexes. Chaff points are generated in a way that guarantee FV security even when the user feature model is compromised.

- b. the user feature indexes  $FI = \{f_i\}_{i=1}^t$  are used to extract feature representation  $F^T = \{f_i^T\}_{i=1}^t$  from the signature template  $T_s$ . This representation is then quantized in  $l/2$ -bits and produces a vector  $Y^T = \{y_i^T\}_{i=1}^t$ .
- c. The features are encoded to produce the locking set  $A = \{a_i\}_{i=1}^t$ , where  $A = X^T || Y^T$ . Hence, the locking elements are represented in a field  $GF(2^l)$ .
- d. the cryptography key  $K$  of size  $KS$  where:

$$KS = l(k + 1) - bits \quad (4.7)$$

is split into  $k+1$  parts of  $l$ -bits each, that constitutes a coefficient vector  $C = \{c_0, c_1, c_2, \dots, c_k\}$ .

A polynomial  $p$  of degree  $k$  is encoded using  $C$ , where  $p(x) = c_k x^k + c_{k-1} x^{k-1} + \dots + c_1 x + c_0$ .



- e. the polynomial is evaluated for all points in  $A = \{a_i\}_{i=1}^t$  and constitutes the set  $p(A) = \{p(a_i)\}_{i=1}^t$  where  $p(a_i) = c_k a_i^k + c_{k-1} a_i^{k-1} + \dots + c_1 a_i + c_0$ .

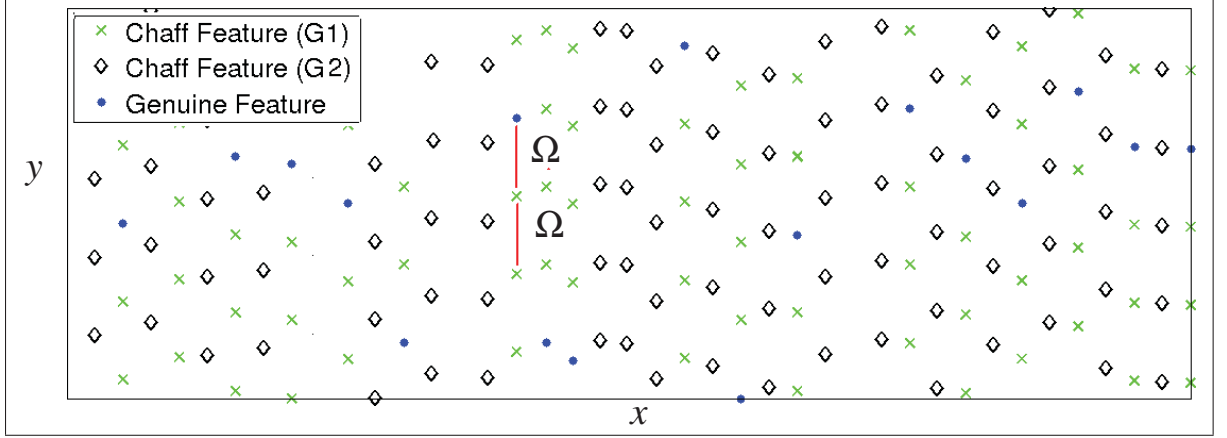


Figure 4.11 Illustration of the chaff generation process: two groups of chaff points are generated. Chaffs of group 1 ( $G_1$ ) have same indexes like that of the genuine points. Chaff and genuine points are equally spaced by a distance  $\Omega$ . Chaffs of group 2 ( $G_2$ ) are generated on different indexes. In case of compromising the genuine indexes, and filtering  $G_2$  out,  $G_1$  still conceal the genuine points from attackers.

- f. chaff (noise) points ( $\hat{A} = \{\hat{a}_{ii}\}_{ii=t+1}^r, \hat{P} = \{\hat{p}_{ii}\}_{ii=t+1}^r$ ) are generated, where  $\hat{a}_{ii} \in GF(2^l), \hat{a}_{ii} \neq a_i \vee ii \in [t+1, r], i \in [1, t]$ , and  $\hat{p}_{ii} \in GF(2^l), \hat{p}_{ii} \neq p(\hat{a}_{ii}) \vee ii \in [t+1, r]$ . The proposed chaff generation method is illustrated in Figure 4.11. A chaff point  $\hat{a}_{ii} = x_{ii} || y_{ii}$  is composed of two parts: the index part  $x_{ii}$  and the value part  $y_{ii}$ . Two groups of chaff points are generated. Chaffs of  $G_1$  have their indexes equal to the indexes of the genuine points. The chaff points and the genuine point that have the same index part are all equally spaced by a distance  $\Omega$ , eliminating the possibility to differentiate between the chaffs and the genuine point. Chaffs of  $G_2$  have their index part differ than that of the genuine points<sup>7</sup>. As the number of chaffs in  $G_1$  is limited by the parameters  $t$  and  $\Omega$ , so to inject higher quantity of chaffs we define  $\alpha$  as a chaff groups ratio, where:

<sup>7</sup>The user password protects the  $UR$  that stores his feature representation model. If the attacker compromised the password, the indexes of the genuine points are known to him. In such case, chaffs of  $G_2$  are filtered out while  $G_1$  could not be filtered without applying good features that are extracted from a genuine signature image. So,  $G_1$  secures the genuine points even if the user password is compromised.

$$\alpha = g_2/g_1 \quad (4.8)$$

where  $g_1$  and  $g_2$  are the amount of chaff features belong to  $G_1$  and  $G_2$ , respectively.  $G_2$  chaffs are generated with  $\alpha t$  indexes different than the  $t$  genuine indexes<sup>8</sup>. Hence, the FV size  $r$  is given by:

$$r = t(\alpha + 1)/\Omega \quad (4.9)$$

So, the total number of chaffs  $z$  is given by:

$$z = t(\alpha + 1 - \Omega)/\Omega \quad (4.10)$$

- g. the genuine set  $(A, p(A))$ , and the chaff set  $(\hat{A}, \hat{P})$  are merged to constitute the fuzzy vault  $FV = (\tilde{A}, \tilde{P})$ , where  $\tilde{A} = A \cup \hat{A}$ ,  $A = \{a_i\}_{i=1}^t$ ,  $\hat{A} = \{\hat{a}_i\}_{i=t+1}^r$  and  $\tilde{P} = p(A) \cup \hat{P}$ ,  $p(A) = \{p(a_i)\}_{i=1}^t$ ,  $\hat{P} = \{\hat{p}_i\}_{i=t+1}^r$ .

#### 4.4.2 Authentication process

The authentication sub-system uses the user query sample  $Q$  and the password  $PW$ , to decode the fuzzy vault  $FV$  and restore the user cryptography key  $K$ . First the password  $PW$  is used to decrypt the  $UR$  matrix. Then the vectors  $FI$ ,  $VI$ , and  $\Delta$  are used to decode the FV by means of the query sample  $Q$ .

The FV decoding module (illustrated in Figure 4.12) describes the following processing steps:

- a. the virtual indexes  $VI = \{vI_i\}_{i=1}^t$  are quantized in  $l/2$ -bits and produces a vector  $X^Q = \{x_i^Q\}_{i=1}^t$ .

---

<sup>8</sup>The total number of indexes that should be mapped in  $VI$  is  $(\alpha+1)t$  indexes. As these indexes are represented in  $l/2$ -bits, so  $(\alpha+1)t < 2^{l/2}$ , i.e.,  $\alpha < ((2^{l/2})/t) - 1$ . E.g., for  $l = 16$ -bits,  $t = 20$ ,  $\alpha \leq 11$ .

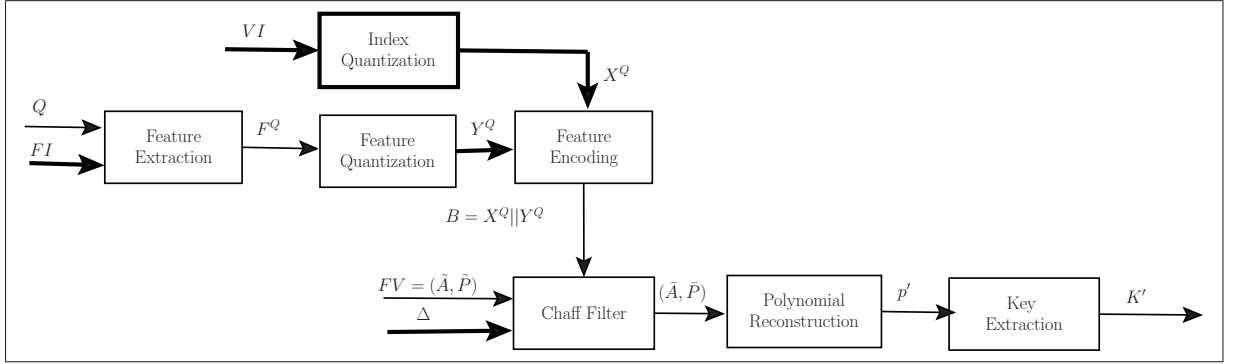


Figure 4.12 Proposed FV Decoding. The bold lines highlight the modules where specific modifications apply to the standard decoding process illustrated in Figure 4.2. A pre-selected user-specific feature model  $FI$  is used for feature extraction and it is mapped to virtual indexes  $VI$ . FV unlocking points are generated by concatenating the feature values to their indexes. Chaff points are filtered by matching FV points with the unlocking points based on expected feature variability  $\Delta$ . Finally, the matching set  $(\bar{A}, \bar{P})$  is used to reconstruct the polynomial  $p'$  and its coefficients constitute the crypto-key  $K'$ .

- b. the user feature indexes  $FI = \{f_i\}_{i=1}^t$  are used to extract feature representation  $F^Q = \{f_i^Q\}_{i=1}^t$  from the query sample  $Q$ . This representation is then quantized in  $l/2$ -bits and produces a vector  $Y^Q = \{y_i^Q\}_{i=1}^t$ .
- c. The features are encoded to produce the unlocking set  $B = \{b_i\}_{i=1}^t$ , where  $B = X^Q || Y^Q$ . Hence, the unlocking elements are represented in a field  $GF(2^l)$ .
- d. the unlocking set  $B$  is used to filter the chaff points from the FV. An adaptive matching window (AMW) method is applied to match unlocking and locking points. Items of  $B$  are matched against all items in  $\tilde{A}$ . This process results in a matching set  $(\bar{A}, \bar{P}) = ((B \cap \tilde{A}), p \leftarrow (B \cap \tilde{A}))$ , where  $p \leftarrow (B \cap \tilde{A})$  represents the projection of the matching features on the polynomial space. The proposed chaff filtering with adaptive matching window (AMW) method is illustrated in Figure 4.13. If the feature indexes are correct<sup>9</sup>, then all elements of  $X^Q$  will have corresponding elements in  $X^T$ . So, all of chaffs of

<sup>9</sup>That occurs if the applied password is genuine, so the  $UR$  is decrypted properly and the right indexes are restored.

$G_2$  will be filtered out (see the left figure). Then, each of the remaining FV points will be compared to corresponding points extracted from the query sample. The AMW method is applied: for every feature  $i$ , a matching window  $w_i$  is adapted to the feature modeled variability  $\delta_i$ , where  $w_i = 2\delta_i$ . A FV point  $a_i$  is considered matching with an unlocking point  $b_i$ , if they reside in the same matching window. I.e.,  $|a_i - b_i| \leq w_i$ .<sup>10</sup>

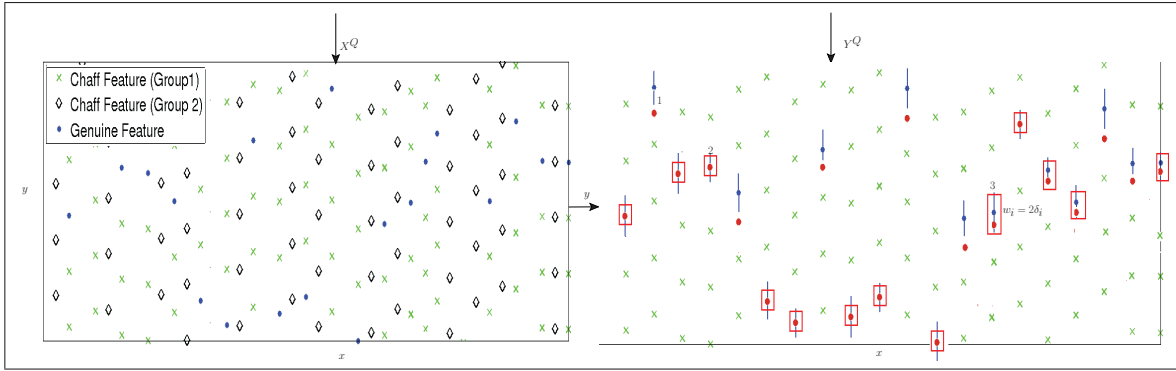


Figure 4.13 Proposed Chaff Filtering with AMW: in the left side, the  $X^Q$  part of unlocking set  $B$  filters  $G_2$  chaffs.  $G_2$  is filtered only when applying correct feature indexes. In the right side, chaffs of  $G_1$  are filtered by matching the remaining FV points with the unlocking points extracted from the query sample. Only the  $Y^Q$  part of  $B$  is needed at this step. For each index  $i$ , an adaptive matching window  $w_i = 2\delta_i$  is applied. For instance, point 1 is not isolated as the point that extracted from the query sample (shown in red color) resides out of the matching window attached to the genuine FV point. On the other hand, the genuine FV point and the query point have exact values for point 2 and they reside in the same matching window for point 3.

- e. the matching set  $(\bar{A}, \bar{P})$  is used to reconstruct a polynomial  $p'$  of degree  $k$  by applying the R-S decoding algorithm (Berlekamp and Elwyn, 1968).
- f. the coefficients of  $p'$  are assembled to constitute the secret cryptography key  $K'$ .

<sup>10</sup>As  $a_i = x_i^T || y_i^T$ ,  $b_i = x_i^Q || y_i^Q$  and  $x_i^T = x_i^Q$ , then the above matching condition implies that  $|y_i^T - y_i^Q| \leq w_i$ .

### 4.4.3 Security analysis

The security of the proposed FV system is analyzed in terms of the brute-force attack. Assume an attacker could compromise the FV without possessing neither valid password nor genuine signature sample. In this case, the attacker tries to separate enough number of genuine points  $(k + 1)$  from the chaff points.

To this end, the  $r$  points of the FV are searched for a correct  $k + 1$  points needed to reconstruct the polynomial  $P$  and retrieve the secret key  $K$ . According to the proposed FV encoding scheme, not all of the FV points must be searched, as among the  $(\alpha + 1)t$  feature indexes used to generate FV genuine and chaff points, only  $t$  of them are genuine indexes. Therefore, attacker must first locate the genuine indexes and then search the points with these indexes. However, locating only  $k + 1$  genuine indexes is enough to filter the  $k + 1$  points needed for polynomial reconstruction. The number of FV points with the same index is  $1/\Omega$ <sup>11</sup>. Along them there is only one genuine point and the other points are chaffs. Hence, the overall security (brute-force search space for an attacker) is given by:

$$security \cong \binom{(\alpha + 1)t}{k + 1} (1/\Omega)^{k+1}. \quad (4.11)$$

- Example: consider a FV of encoding size ( $t = 20$ ), a polynomial degree  $k = 7$  (for encoding a secret key  $K$  of length of 128-bits), a chaff separation parameters  $\Omega = 0.2$ , and chaff groups ratio  $\alpha = 1$ . So, the search space is  $\cong \binom{2t}{k+1} (1/\Omega)^{k+1} = \binom{40}{8} \times 5^8 \cong 2^{45}$ . So, the entropy of a FV system with such parameters is equivalent to 45 bits.

An attacker may decide to guess the password and use it to decrypt  $EUR$  (to locate the genuine indexes and filter  $G_2$ ). Then, he searches the remaining points to filter  $G_1$  and isolate the matching set  $(\bar{A}, \bar{B})$ . In such case, the search space is narrowed to  $5^8 \cong 2^{18}$ . However, an eight

---

<sup>11</sup>The parameter  $\Omega$  should be properly selected so that  $1/\Omega$  is an integer

character password has entropy of 18-30 bits (Burr *et al.*, 2006). So, the overall system entropy is about 36-48 bits. Accordingly, in all cases, the proposed bio-cryptographic system provides higher security against brute-force attacks than that of the password-protected cryptography systems.

According to Eq. 4.11, entropy of the system can be increased through using different values of the parameters:  $t$ ,  $\alpha$ ,  $\Omega$ , and  $k$ . While, according to Eq. 4.9 there is a trade-off between FV security and its size. Also, there is trade-off between system security and its recognition accuracy (according to the experiments presented in the following section).

#### 4.5 Experimental results

Feasibility of the proposed system is investigated through adopting two sets of experiments. The first set aims to investigate the ability of the proposed feature representation approach to produce a high quality feature representation. To this end, the impact of each step of feature representation extraction process (as illustrated in section 4.3.) is investigated. In the second set of experiments, the FV recognition performance is investigated. FVs are encoded and decoded (as illustrated in section 4.4) using the extracted feature representation, and then the FV recognition rates are computed. As there is no FV based on offline signature images found in the literature, so we compare our system to the state of the art classical offline signature verification systems SV<sup>12</sup>. As our system relies on two authentication measures (user password and the signature image), and in order to be fair when comparing the system performance to other systems in the literature, we assume in all the experiments (unless the reverse is clearly stated) that the password is compromised by the attacker. In such case, the system performance depends only on the signature query sample (Rathgeb and Uhl, 2010).

---

<sup>12</sup>The BFS technique proposed in this chapter extends the methodologies used to design the WI-SV system (Rivard *et al.*, 2013) and the WD-SV system (Eskander *et al.*, 2012). Therefore, we considered these systems as baseline to our system.

## 4.5.1 Experimental methodology

### 4.5.1.1 Database

The Brazilian database (Freitas *et al.*, 2000) is used for proof-of-concept simulations. It contains 7,920 samples of signatures that were digitized as 8-bit grayscale images over 400X1000 pixels at resolution of 300 dpi. This DB contains three types of signature forgery: random, simple and simulated. For random forgery, the forger does not know neither the signer's name nor the signature morphology. It can also happen when a genuine signature presented to the system is mislabeled to another user. For simple forgery, the forger knows the writer's name but not the signature morphology. He can only produce a simple forgery using a style of writing of his liking. For the simulated forgery, the forger has access to a sample of the signature. He can therefore imitate the genuine signature.

The signatures were provided by 168 writers and are organized as follows: the first 60 writers have 40 genuine signatures, 10 simple forgeries and 10 simulated forgeries per writer, and the other 108 have only 40 genuine signatures per writer. The experimental database is split into two sets: the population-based dataset ( $P$ ) composed of the last 108 writers. The user-based dataset ( $U$ ) composed of the first 60 writers. Set  $P$  is used for the population-based feature selection process. Set  $U$  is split into two subsets: the reference subset ( $R$ ) contains the first 30 genuine signatures, and the questioned subset ( $Q$ ) contains the rest 10 genuine samples, 10 simple and 10 simulated forgeries. The subset  $R$  is used for the user-based feature selection process and both subsets of  $U$  are used for evaluating the system performance.

### 4.5.1.2 Feature extraction

Many different techniques are available to extract features in offline signature images (Impe-dovo and Pirlo, 2008). Any combination of these features may be concatenated into a single high-dimensional representation, and used for the proposed framework. However, we focused

on using feature extracted using extended-shadow-code (ESC) (Sabourin and Genest, 1994), and directional probability density function (DPDF) (Drouhard *et al.*, 1996). Features are extracted based on different grid scales, hence a range of details are detected in the signature image. These features have shown complementary functionality: while ESC detects the spatial information, the DPDF detects the directional information from signature images (Rivard *et al.*, 2013). A set of 30 grid scales is used for each feature type, producing 60 different single scale feature representations. These representations are then fused to produce a feature representation of huge dimensionality ( $M = 30, 201$ ) (Rivard *et al.*, 2013).

#### 4.5.1.3 Feature selection

The two-step process for selection of feature representation is implemented as illustrated in section 4.3. First, population-based dataset ( $P$ ) is used for the population-based feature selection phase. We followed the same experimental settings as in the baseline system in (Rivard *et al.*, 2013). This phase produced a universal ( $PR$ ) representation of dimensionality  $L = 555$ . Second, the user-based dataset ( $U$ ) is used for the user-based feature selection phase. For each user in  $U$ , the signatures in the reference subset  $R$  are used to represent the genuine class and some signatures from the population-based data set  $P$  are used to represent the forgery class. Then, signatures of both classes are represented in the  $PR$  space of  $L$  dimensionality. Finally, the user-based feature selection step runs for  $t$  boosting iterations to learn the most user-specific discriminant features  $FI$ , along with their expected variability  $\Delta$ .

#### 4.5.1.4 FV parameter values

Unless different values are explicitly stated, the FV parameters are set according to Table 4.1. Size of the cryptographic key  $KS$  and chaff separation distance  $\Omega$  have direct impact on the FV security and its memory requirements (see Equations 4.11 and 4.9, respectively). We changed the values of these parameters through some experiments, to test their impact on the system



Table 4.1 FV parameter values

| Parameter                               | Default Value |
|---|---------------|
| Size of the cryptographic key, $KS$     | 128-bits      |
| Chaff separation distance, $\Omega$     | 0.2           |
| No. of genuine points in the FV, $t$    | 20            |
| Chaff group ratio, $\alpha$             | 1             |
| Quantization size of the FV points, $l$ | 16-bits       |
| Decision threshold, $D\Theta$           | 14            |
| Degree of the encoding polynomial, $k$  | 7             |
| No. of FV points, $r$                   | 200           |
| No. of chaff points in the FV, $z$      | 180           |

accuracy. In practical applications, values of these parameters are determined to compromise between accuracy, security and memory complexity. The user-based BFS process showed saturation with 100 boosting iterations. However, we found that, generally, the first twenty features are the most discriminant features. Accordingly, the number of genuine FV points  $t$  is set to 20. Also, the chaff group ratio  $\alpha$  had no impact on the performance. For a good compromise between security and complexity, we set  $\alpha = 1$ . Finally, the quantization size of the FV points  $l$  is set to 16-bits, so that all operations are done in a Galois field ( $GF(2^{16})$ ). This field enables encoding of strong cryptographic keys, e.g., 128-bit keys. The parameters  $D\Theta$ ,  $k$ ,  $r$  and  $z$  are not set directly, but they are computed according to Equations 4.2, 4.7, 4.9 and 4.10, respectively.

#### 4.5.1.5 Performance measures

To investigate the quality of feature representation, similarity scores are computed<sup>13</sup>, and used to generate a ROC curve for each user in the dataset  $U$ . A ROC curve plots the False Accept Rate (FAR) against the Genuine Accept Rate (GAR) for all possible matching thresholds (all

<sup>13</sup>The power of the proposed technique for feature representation relies on two concepts: the first is the selection of a concise user-specific representation through a two-step BFS process, and the second is to learn the representation variability and use it for matching FV points adaptively. To investigate the impact of each of these concepts separately, we compute the similarity scores  $SS^Q$  twice: first according to Eq.4.1 to investigate the first concept, and second according to Eq.4.4 to investigate the second concept.

generated scores). FAR for a specific threshold is the ratio of forgery query samples that have a number of matching encoded features exceeds this threshold. GAR is the ratio of genuine query samples that have a number of matching features greater than the threshold.

In order to have a global assessment on the quality of feature representation over all users in  $U$ , we compute and average the area under the ROC curve (AUC) for all users. High AUC indicates more separation between the similarity score distributions of the genuine and forgeries classes.

To assess the FV recognition performance, the average error rate ( $AER_{all}$ ) is computed, where

$$AER_{all} = (FRR + FAR_{random} + FAR_{simple} + FAR_{simulated})/4 \quad (4.12)$$

False Reject Rate ( $FRR$ ) is the ratio of genuine queries that failed to decode the FV,  $FAR_{random}$ ,  $FAR_{simple}$  and  $FAR_{simulated}$  are the ratio of random, simple, and simulated forgeries respectively that succeed to compromise the system and decode the FV.

In literature, recognition performance of bio-cryptographic systems are mostly tested using forgery samples that are belonging to other users. This case is equivalent to testing our system against random forgeries only. Accordingly, for having a fair comparison with other systems, we also test the resistance of the system against each type of forgeries separately. To this end, the following performance measures are also reported:

$$AER_{random} = (FRR + FAR_{random})/2 \quad (4.13)$$

$$AER_{simple} = (FRR + FAR_{simple})/2 \quad (4.14)$$

$$AER_{simulated} = (FRR + FAR_{simulated})/2 \quad (4.15)$$

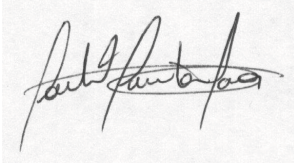
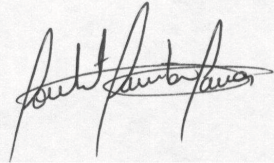
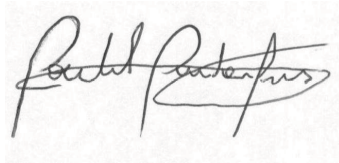
#### 4.5.2 Results on quality of feature representation

The user-based dataset  $U$  is used to evaluate the approach proposed in Section 4.3 for feature representation selection. To this end, for each user, the 30 signatures in the reference set  $R$  are matched with the 40 signatures in the query set  $Q$ . This experiment results in 1200 matchings per user, of them 300 with genuine samples, and 900 with forgery samples (300 random, 300 simple, and 300 simulated forgeries). The  $t$  user-based features are extracted from each sample and used for producing FV encoded features. For encoding, both the feature index and the feature values are quantized in  $l/2$ -bits and then concatenated to produce a  $l$ -bit encoded feature. The encoded features produced from the reference signatures constitutes a FV locking set and concealed with chaff points. The encoded features produced from the query signatures constitutes the FV unlocking set. Matching the points of both the reference and query samples produces a similarity score (ranging from 0 to  $t$ ). Point matching is either done based on Eq.4.1 or based on Eq.4.4. In the later case, a matching window is adapted to the modeled feature variability ( $mw_i = 2\delta_i$ ), where two points are considered similar if they reside in the corresponding matching window.

To clarify the aforementioned experimentation, we investigate the result of such matching for the first user in the dataset  $U$ . Table 4.2 shows a template signature of this user along with a genuine and a simulated forgery query samples. Figure 4.14 shows the matching process in the feature encoding space. In this illustrative example, the feasibility of using the proposed feature representation technique is clear. The feature representation is discriminant as while the simulated forgery query  $Q_2$  resembles the genuine sample  $Q_1$ , their similarity scores differs much. For instance,  $SS_1^Q = 18$  and  $SS_2^Q = 7$ . For this example, setting the FV parameters so that  $D\Theta \in [8, 18]$  results in acceptance of the genuine sample and rejection of the forgery sample. Also, the impact of applying the AMW approach is clear. For the genuine query  $Q_1$ , the number of features that have exact values for both the template and query images is 11

features (i.e., when employing Eq. 4.1). While, this value is increased to 18 features when employing the AMW approach (i.e., when employing Eq. 4.4).

Table 4.2 Signature samples of the first user in dataset  $U$

| Template T  | Genuine Query $Q_1$   | Simulated Forgery Query $Q_2$  |
|---|---|--|
|  |  |  |

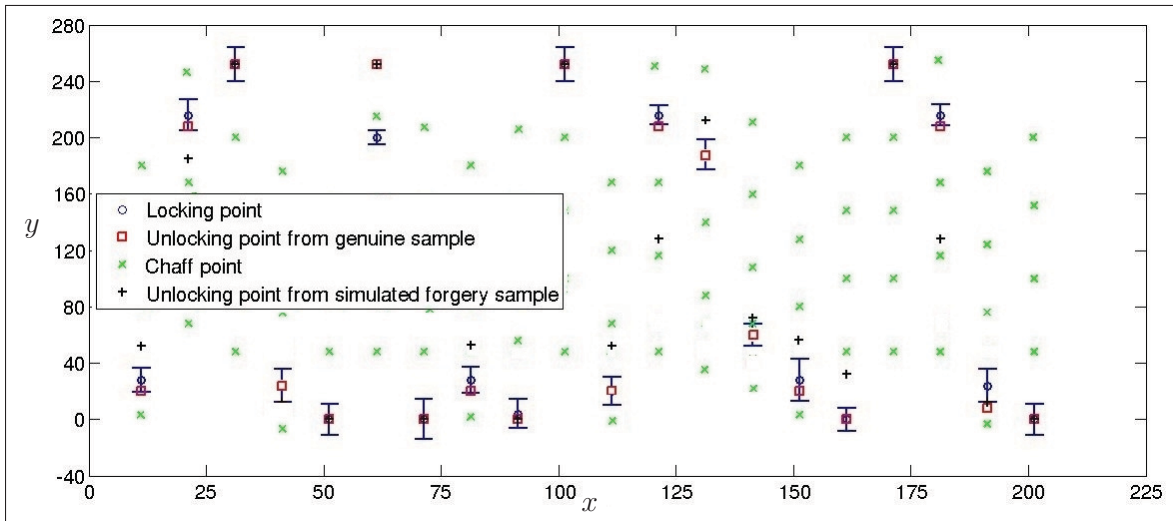


Figure 4.14 Illustration of chaff filtering in feature encoding space for the first user in the dataset  $U$ : out of the  $t = 20$  encoded FV points, only 11 from the genuine unlocking points have exact value as their corresponding locking points (according to Eq.4.1,  $SS_1^Q = 11$ ). Applying the AMW method increased the similarity score (according to Eq.4.4,  $SS_1^Q = 18$ ). For the simulated forgery sample,  $SS_2^Q = 7$ .

The aforementioned example is a result of only two matchings (one with a genuine sample and another with a simulated forgery sample). However, the employed experimental protocol consists in 1200 matchings per user. In order to have a global assessment for all matchings for this specific user, the similarity score distribution is computed for all the matchings. Also, to investigate the impact of applying each step of the feature representation extraction process,

four scenarios of feature representation are implemented to generate and match the FV locking and unlocking points. In the scenarios, different feature vectors are encoded in a field  $GF(2^l)$  as follows:

- a. random features:  $t$  features are randomly selected from the  $D$  features of the multi-feature representation  $M$ .
- b. population-based features: the most discriminant  $t$  features of  $PR$  representation (of dimension  $L$ ) are used.
- c. user-based features: the  $t$  features embedded in the user-specific feature indexes vector  $UR$  are used. For this and the above scenarios, Eq.4.1 is applied to compute the similarity scores.
- d. user-based features with adaptive matching window: the feature vector is the same as of the above scenario, while the AMW is employed when computing the similarity scores (i.e., Eq. 4.4 is applied).

Figure 4.15 shows the similarity score distribution for all of scenarios. It is clear that, for this user, each step of the proposed approach for feature selection enhanced the quality of the feature representation. Accordingly, using the extracted feature representation for FV locking and unlocking is expected to produce high recognition rates. However, the actual recognition rate depends on the selected score threshold that split the genuine and forgery classes. For instance, in the last scenario of Fig 4.15, setting the FV decoding threshold  $D\Theta = 14$  results in  $FRR = 0$  with small  $FAR$ . On the other hand, when  $D\Theta = 15$  the  $FAR = 0$  with small  $FRR$ .

In order to investigate the expected recognition rates for the different thresholds, the similarity scores are used to generate ROC curves. Figure 4.16 shows the ROC curve for the first user in the dataset  $U$ .

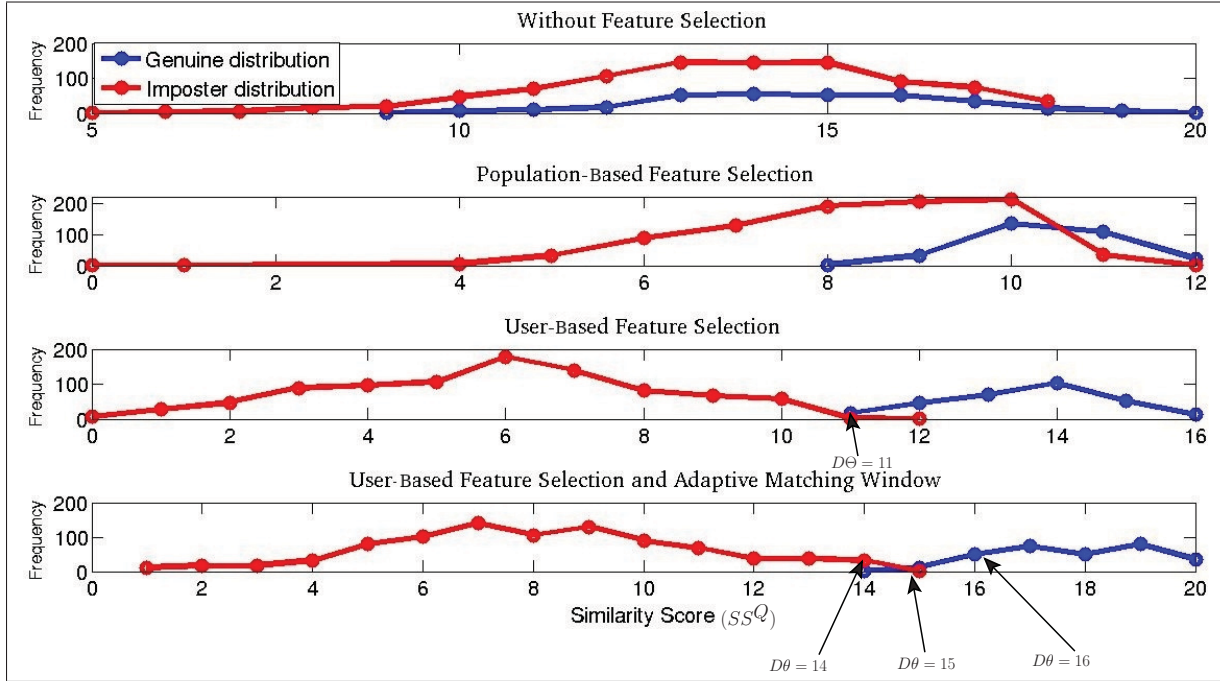


Figure 4.15 Similarity score distribution for the first user in dataset  $U$ : feasibility of the proposed BFS is clear. Each step of the proposed technique increases the separation between the similarity distributions of the genuine and forgery classes. Applying the AMW method, even it did not impact the class separation for this specific user, it increased the range of similarity scores. Increasing this range made it possible to select feasible decoding thresholds. For instance, without adapting the matching window, a good compromise between  $FRR$  and  $FAR$  is to set  $D\Theta = 11$  (see the 3<sup>rd</sup> scenario). In this case,  $k = 1$ , and  $KS = 32$ -bits, which is not feasible (see Eq.4.2 and Eq.4.7, respectively). On the other hand, when AMW is employed, a good threshold setting is  $D\Theta = 14$  (see the 4<sup>th</sup> scenario). In this case,  $k = 7$  and  $KS = 128$ -bits, which is a feasible setting.

Table 4.3 Average AUC over the 60 users in the dataset  $U$  for the different design scenarios

| Design Aspect | Without Feature Selection | Population-based Feature Selection | User-based Feature Selection | User-based Feature selection with AMW |
|---------------|---------------------------|------------------------------------|------------------------------|---------------------------------------|
| Average AUC   | 0.6577                    | 0.7724                             | 0.9328                       | 0.9700                                |

The above analysis is shown only for the first user in the experimental dataset  $U$ . However, the impact of each step of the proposed feature selection approach differs for the different users.

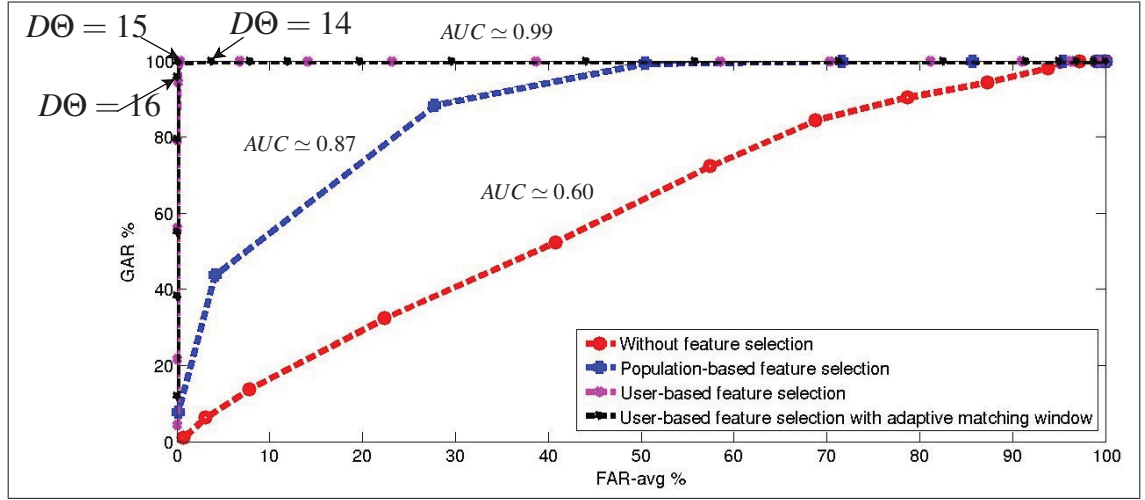


Figure 4.16 ROC Curve for the first user in the dataset  $U$ : each of the BFS steps increased the AUC. For the classical classifiers, the operating point could be set directly by setting the classifier threshold. However, for the FV systems, the operating point are equivalent the decoding threshold  $D\Theta$ . Considering a FV with parameters as shown in Table 4.1, the three operating points showed in Figure 4.15 and in Figure 4.16 ( $D\Theta = 14, 15, 16$ ) are realized by employing encoding polynomials with degree  $k = 7, 9$  and 11, respectively (see Eq.4.2).

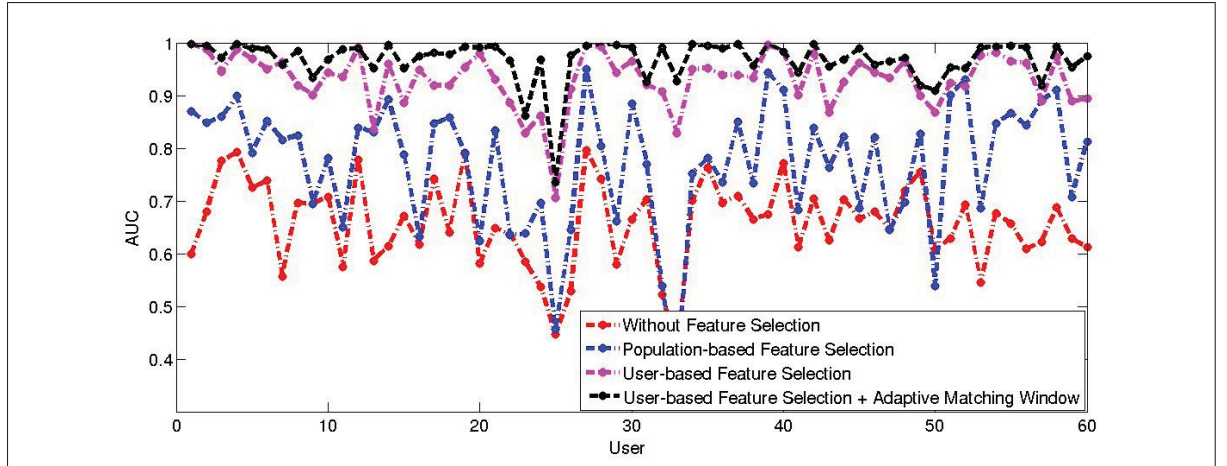


Figure 4.17 AUC for the 60 Users in the dataset  $U$ : impact of each step of the proposed BFS technique differs for the different users. While, generally, the representation quality is enhanced with each step.

In order to have a global quantitative assessment, we compute the area under the ROC curve (AUC) for all users and for all scenarios. High AUC indicates more separation between the



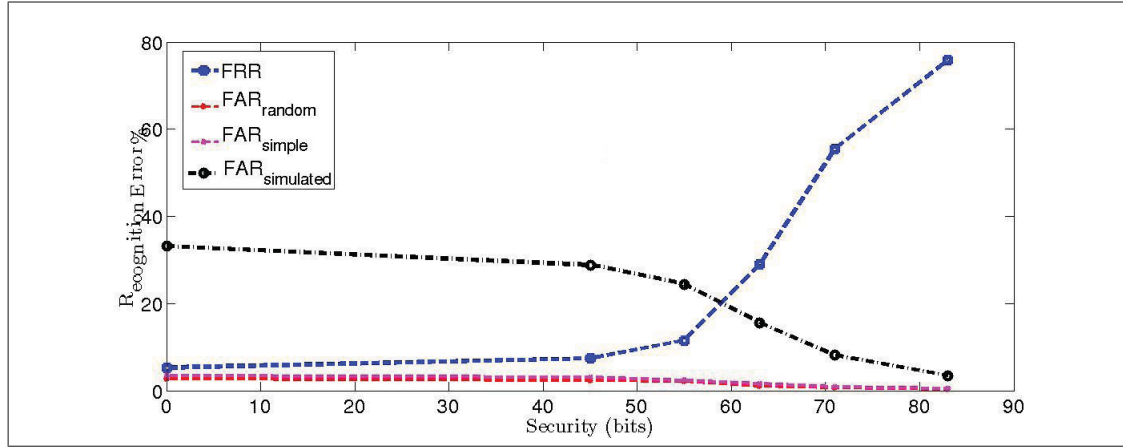


Figure 4.18 Trade-off between FV security and recognition measures.

similarity score distributions of the genuine and forgeries classes. Table 4.3 shows the average AUC for all users for the different design aspects. Figure 4.17 shows the AUC values for all users for the different design scenarios. It is obvious that, generally, each step in the proposed approach enhances the representation quality. However, the impact of each step depends on the nature of the template and the forgery signatures. Generally, the produced feature representation is discriminant, as for 54 out of 60 investigated users, the  $AUC > 95$ . Only two users have  $AUC < 90$ : user 23 with  $AUC \simeq 85$  and user 25 with  $AUC \simeq 75$ . Future work will investigate the performance for the different individuals based on the Biometric Menagerie concept (Yager and Dunstone, 2010), and user-specific design issues maybe proposed to guarantee acceptable performance for any user.

Table 4.4 FV performance for different key sizes

| KS                       | 128-bits | 160-bits | 192-bits |
|--------------------------|----------|----------|----------|
| k                        | 7        | 9        | 11       |
| DΘ                       | 14       | 15       | 16       |
| FRR                      | 11.53    | 14.05    | 20.68    |
| FAR <sub>random</sub>    | 2.05     | 1.5      | 0.93     |
| FAR <sub>simple</sub>    | 2.39     | 1.93     | 1.41     |
| FAR <sub>simulated</sub> | 24.38    | 20.07    | 15.14    |
| AER <sub>all</sub>       | 10.08    | 9.38     | 9.54     |



Table 4.5 Impact of chaff quantity on the FV performance

| Chaff separation ( $\Omega$ )   | Without chaff | 0.2     | 0.10    | 0.05    | 0.025   |
|---------------------------------|---------------|---------|---------|---------|---------|
| No. of FV points ( $r$ )        | 20            | 200     | 400     | 800     | 1600    |
| No. of chaff points ( $z$ )     | 0             | 180     | 380     | 780     | 1580    |
| Security                        | 0-bits        | 45-bits | 52-bits | 60-bits | 68-bits |
| FRR                             | 5.25          | 11.53   | 28.94   | 55.53   | 75.81   |
| $\text{FAR}_{\text{random}}$    | 2.74          | 2.05    | 1.06    | 0.58    | 0.31    |
| $\text{FAR}_{\text{simple}}$    | 3.49          | 2.39    | 1.58    | 0.88    | 0.49    |
| $\text{FAR}_{\text{simulated}}$ | 33.14         | 24.38   | 15.63   | 8.15    | 3.42    |
| $\text{AER}_{\text{all}}$       | 11.15         | 10.08   | 11.80   | 16.28   | 20.00   |
| $\text{AER}_{\text{random}}$    | 3.99          | 6.79    | 15      | 28.05   | 38.06   |
| $\text{AER}_{\text{simple}}$    | 4.37          | 6.96    | 15.26   | 28.20   | 38.15   |
| $\text{AER}_{\text{simulated}}$ | 19.19         | 17.95   | 22.28   | 31.84   | 39.61   |

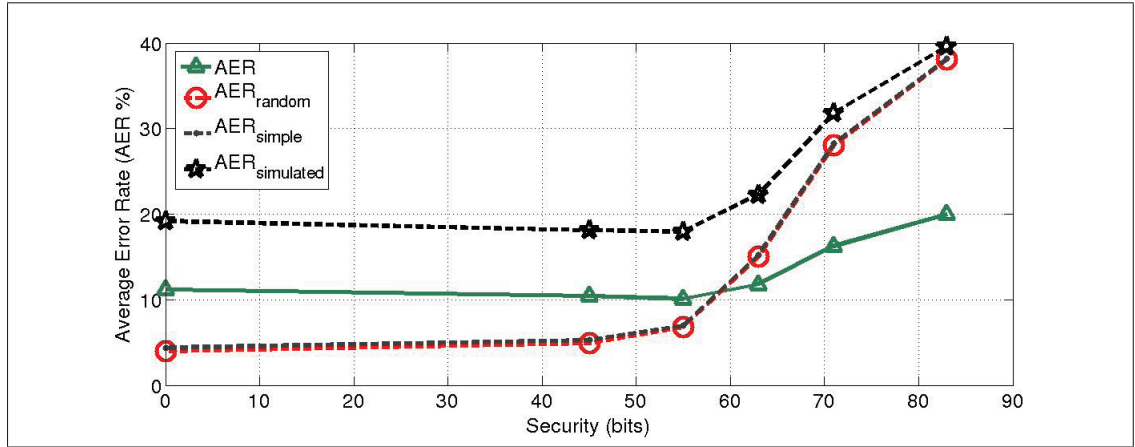


Figure 4.19 Trade-off between FV security and average recognition error rates.

#### 4.5.3 Results on performance of the FV system

In the following experiments, the performance of the FV system proposed in Section 4.4 is investigated. To this end, FV systems are encoded using the extracted  $UR$  feature representations. The FV parameters was set according to Tabel 4.1, however, some parameters are changed through the experiments to test their impact on the system security and recognition performance.

To test the impact of the crypto-key size ( $KS$ ) on the FV recognition performance, we encode FVs with keys of different sizes as shown in Table 4.4. The parameters  $D\Theta$  and  $k$  are computed according to Eq.4.2 and Eq.4.7, respectively. Changing the key size  $KS$  have shown a trade-off between the genuine recognition and the false acceptance rates. While, encoding larger (more secure) cryptographic keys decreases the  $FAR$ , it increases the  $FRR$ .

To test the impact of quantity of chaff points on the FV recognition performance, we investigate different values of the chaff separation parameter  $\Omega$  as shown in Table 4.5. The FV size  $r$ , the chaff size  $z$  and system security are computed according to Eq.4.9, Eq.4.10 and Eq.4.11, respectively. It is obvious that the chaff quantity, and hence, the FV size increases when smaller values of the parameter  $\Omega$  are used. While, there is a trade-off between the system security and its recognition performance.

Figures 4.18, 4.19 illustrate the impact of the quantity of chaffs on both the security and the recognition quality. As shown in Figure 4.18, injecting more chaffs increases the FV security and its resistance to the false acceptance, while the genuine acceptance rate is degraded. It is important to note that impact of chaff quantity differ for the different type of forgeries. While the  $FAR$  for both random and simple forgeries is not much decreased, the system resistance to incorrectly accept simulates forgeries is much decreased. Figure 4.19 shows that the Average Error Rates ( $AER$ ), that are computed according to Eq.4.12- Eq.4.15, are increasing with increasing chaff quantity. So, there is a trade-off between system security and its recognition performance. Also, It is obvious that the performance is better when only the random forgeries are considered ( $AER_{random} \simeq 6.8\%$  with 45-bits security).

Through all of the above experiments, we assumed that the user password is compromised. However, to report the actual performance of the system we have to consider the case when an attacker neither possesses a correct password nor a genuine signature sample. In this case, he can not decrypt the  $UR$  model and hence he randomly guess the feature indexes  $X^Q$ . Table 4.6

Table 4.6 Impact of using a user password as a second authentication measure

| Measure           | Compromised Password | Secured Password |
|-------------------|----------------------|------------------|
| $FRR$             | 11.53                | 11.53            |
| $FAR_{random}$    | 2.05                 | 0                |
| $FAR_{simple}$    | 2.39                 | 0                |
| $FAR_{simulated}$ | 24.28                | 0                |
| $AER_{all}$       | 10.08                | 2.88             |

shows the recognition performance in such case. No impostor query was incorrectly accepted and the overall  $AER$  is significantly decreased (the recognition accuracy is about 97% ).

As there is no FV based on offline signature images found in the literature, so we compare our system to the state of the art classical offline signature verification systems SV. Tabel 4.7 compares the performance of the proposed FV system with the baseline WI-SV (Rivard *et al.*, 2013) and WD-SV (Eskander *et al.*, 2012) systems. The majority vote concept is tested to fuse decisions of multiple FVs (each FV encoded by features extracted from a single sample). Ensembles of 1,3,5,7,9,11,13 and 15 FVs are tried. It is shown that using multiple FVs enhanced the performance.

To be fair when comparing the proposed two-factor authentication system, that uses a signature and a password, with the classical SV systems, we assume here that the password is compromised by the attacker. In this case, the proposed FV has shown less recognition performance than both of the classical SV systems. However, if the password is secure, the proposed FV system outperforms the other systems, as the  $AER_{all} = 2.88$ . Moreover, the FV system is the only system that can be employed to enforce authenticity of the cryptography schemes like encryption and digital signatures. Also, the classification label of both of the classical SV system can be bypassed (Uludag, 2006), however, it is trusted with the FV system as it results from a protected decoding process. The confidentiality of the signature templates is not achieved with the WI-SV systems, while it is perfectly achieved with the WD-SV as no signature references are needed for the classification decision. The template security of the FV system depends on

Table 4.7 Performance of the baseline SV systems and the proposed FV system

| System                                | Measure %         | Number of signature templates |       |       |       |         |         |         |         |  |  |
|---------------------------------------|-------------------|-------------------------------|-------|-------|-------|---------|---------|---------|---------|--|--|
|                                       |                   | 1                             | 3     | 5     | 7     | 9       | 11      | 13      | 15      |  |  |
| WI-SV (Rivard <i>et al.</i> , 2013)   | $FRR$             | 13.53                         | 10.66 | 9.83  | 9.75  | 9.36    | 9.69    | 9.80    | 9.77    |  |  |
|                                       | $FAR_{random}$    | 0.12                          | 0.06  | 0.04  | 0.03  | 0.03    | 0.02    | 0.03    | 0.02    |  |  |
|                                       | $FAR_{simple}$    | 0.43                          | 0.33  | 0.32  | 0.33  | 0.32    | 0.33    | 0.32    | 0.32    |  |  |
|                                       | $FAR_{simulated}$ | 14.95                         | 12.52 | 11.87 | 11.36 | 11.55   | 11.11   | 10.77   | 10.65   |  |  |
|                                       | $AER_{all}$       | 7.26                          | 5.89  | 5.52  | 5.37  | 5.32    | 5.29    | 5.23    | 5.19    |  |  |
| WD-SV (Eskander <i>et al.</i> , 2012) | $FRR$             | 7.83                          |       |       |       |         |         |         |         |  |  |
|                                       | $FAR_{random}$    | 0.016                         |       |       |       |         |         |         |         |  |  |
|                                       | $FAR_{simple}$    | 0.17                          |       |       |       |         |         |         |         |  |  |
|                                       | $FAR_{simulated}$ | 13.50                         |       |       |       |         |         |         |         |  |  |
|                                       | $AER_{all}$       | 5.38                          |       |       |       |         |         |         |         |  |  |
| FV                                    | $FRR$             | 11.53                         | 9.07  | 8.53  | 8.19  | 7.95    | 7.84    | 7.80    | 7.69    |  |  |
|                                       | $FAR_{random}$    | 2.05                          | 0.44  | 0.13  | 0.040 | 1.6E-04 | 1.5e-04 | 1.0e-04 | 5.0e-05 |  |  |
|                                       | $FAR_{simple}$    | 2.39                          | 2.06  | 2.01  | 1.97  | 1.89    | 1.95    | 1.92    | 1.92    |  |  |
|                                       | $FAR_{simulated}$ | 24.38                         | 23.33 | 23.26 | 22.98 | 23.22   | 23.04   | 23.09   | 23.26   |  |  |
|                                       | $AER_{all}$       | 10.08                         | 8.72  | 8.48  | 8.29  | 8.26    | 8.20    | 8.20    | 8.21    |  |  |

Table 4.8 Comparison of the baseline SV Systems and the proposed FV system

| System                                 | Recognition rate | Cryptography applicability | Template security | Trusted label | Computational complexity |
|--|------------------|----------------------------|-------------------|---------------|--------------------------|
| WI-SV<br>(Rivard <i>et al.</i> , 2013) | high             | no                         | no                | no            | medium                   |
| WD-SV<br>(Rivard <i>et al.</i> , 2013) | high             | no                         | yes               | no            | low                      |
| Proposed FV                            | medium           | yes                        | yes               | yes           | high                     |

the amount of chaff points. Concerning the computational complexity (for the authentication mode), the WD-SV is the most light system as it relies on concise feature representation and a single classification step. On the other hand, the WI-SV system used large amount of features and fused multiple classification labels for enhanced recognition performance. The FV system, although uses a concise feature representation, sufficient amount of chaff points must be added for its security. Also, the FV system employs polynomial decoders that significantly increase its computational complexity. Table 4.8 summarizes the aforementioned comparison between the proposed FV and the baseline systems.

#### 4.5.4 Computational complexity

The proposed system consists of two processes: 1) enrollment, and 2) authentication (see Figure 4.9). For the complexity of the enrollment process, the user-representation selection module, and more specifically, the two-step BFS (see Figure 4.8), is the most time consuming module. According to Rivard *et al.* (Rivard *et al.*, 2013), the time complexity of the BFS algorithm is  $O(DST)$ , where  $D$  is the dimensionality of the initially extracted multi-feature representation,  $S$  is the number of training samples, and  $T$  is the number of boosting iterations. This analysis indicates high computational growth rate for high dimensional problems, like that for the population-based BFS step. However, the computational growth rate is much slower for the user-based BFS, where dimensionality of the representation is significantly reduced through the first step. Moreover, smaller number of user-specific training samples and

less boosting iterations are required for producing a concise user-specific representation that encodes the FV.

On a Linux based server of 24 GB memory size and 8 cores CPU of speed 2.27GHz, the computational times for the different subprocesses are as follows: the population-based BFS takes about 2 days, however, this step is done only once for the whole system. To enroll a new user to the system, the following subprocesses are executed: multi-feature extraction (14 s /image), user-based BFS (120 s /user), and FV encoding (1 s/ FV). So, for a relatively large number of training images, e.g., 30 images/user, a user enrollment took about 541 s (9 min). Accordingly, the enrollment phase of the proposed method is relatively complex due to the employed representation selection process. The authentication process implies a much less computational complexity. On the same processing environment, the total authentication time is about 14.25 s (14 s for feature extraction + 0.25 sec for FV decoding). This is comparable with state-of-the-art systems. For instance, a Fingerprint FV proposed by Nandakumar et al. (Nandakumar *et al.*, 2007), with decoding time of about (8 s).

#### **4.6 Conclusions and future work**

In this chapter a FV system based on offline signature images is proposed. A new two-step boosting feature selection method is proposed, and used to select good features while learning their variations for each specific user. It is shown that selecting features based on simulated signatures could represent the actual system users. While, running another user-specific feature selection process enhanced the quality of feature representation. Also, adapting the features matching window based on their expected variations results in better FV performance. A user password is used as a second authentication measure to enhance FV system accuracy. To enhance the FV recognition performance, a simple ensemble of FVs is produced by applying the majority vote decision fusion concept.

The proposed FV implementation can be applied to alleviate the key management problem within cryptographic schemes like encryption and digital signature. Moreover, it can be employed as a secure signature verification (SV) scheme. In this case, it provides template protection, as the signature templates are stored in an encrypted form within the FV template. Also, this SV scheme produces trusted classification decisions, as the cryptographic key is decoded through a protected mechanism. With the default system parameters (see Table 4.1), brute-force attempts to extract either the signature template or the cryptographic key from the FV require up to  $2^{45}$  trials. Moreover, the proposed scheme facilitates signature revocability, as if the user signature is compromised, different set of features can be extracted from the same signature and produce a new FV template.

Future work shall investigate enhancing different modules of the proposed system. At the feature extraction level, fusion of more different types of features may enhance the feature representation. For the feature selection module, it might be useful to represent knowledge about the simulated forgeries by using such samples in the population-based and/or user-based feature selection phases. For the FV Encoding/Decoding module, some of the FV parameters may be adapted for each user, and the chaff points may be generated adaptively based on the expected feature variations. More advanced ensemble methods may be applied to fuse outputs of single FVs in either score or decision levels. This also motivates designing parallel processing based FV decoders to reduce the decoding complexity. Although applied on offline signatures, the proposed system is generic so can be investigated on other biometrics. Application to multi-modal can also be investigated on either feature, score and decision levels of fusion.

#### **4.7 Discussion**

The proposed FV implementation based on the offline signature images demonstrated reliable performance. However, a trade-off between FV accuracy and security has been shown. Embed-

ding of high number of chaffs increases its entropy against brute-force attack, while it impacts the recognition accuracy as chaffs interfere with genuine encoding points. In Appendix II, an adaptive chaff generation method is proposed, where the feature variability modeled in the DR space controls the chaff generation process. This way, higher chaff embedding does not impact much the accuracy. The proposed implementation can be employed to design handwritten signature based encryption and digital signature schemes. In Appendix III, a realization of digital signatures with offline handwritten signatures is proposed. Through this method, integrity of documents can be achieved, while users continually employing the traditional handwritten signatures.



## GENERAL CONCLUSION

In this Thesis methods to design offline signature verification (OLSV) and bio-cryptography systems based on the offline signature images are investigated. The main objective of the research conducted was to develop systems that enable automating paper-based processes, while migrating to electronic systems is transparent to users. For instance, a user does not need to carry a smart card to authenticate his transactions, to access his confidential information, or to sign his transactions digitally. A user, instead, continues to sign the traditional documents by hand and to use the existing paper-based forms, that are scanned and integrated in the electronic systems. The offline signature images are authenticated through employing OLSV systems, and also it could be used to enforce confidentiality and integrity through employing the offline signature-based bio-cryptographic technique.

Limitations of the feature representation (FR) design approaches are alleviating by employing the dissimilarity representation (DR) approach, where dissimilarities among signature images constitute the classification space. We proposed a FR-based DR approach, where the DR is built on top of a traditional FR. An optimization approach is proposed that produces a concise and discriminant DR from huge number of feature extractions and prototypes. The OLSV classifiers and the bio-cryptographic FV scheme are formulated as a thresholding classifiers in the optimized DR space. Complete implementations of these systems are presented, and simulation results have demonstrated their viability. Validating the proposed DR approach on the offline signature biometrics motivates future investigations of similar techniques for other biometric and pattern recognition problems.

In the first contribution (Chapter II) class-independent DRs are produced by running a boosting feature selection (BFS) algorithm in a DR space, which we called feature-dissimilarity (FD) space where distances between individual features are the space constituents. This FD space is tuned to specific classes using their training data producing sparser and more discriminant

class-specific FD space. For selecting efficient prototypes for each class, the produced FD space is translated to another DR space which we call dissimilarity (D) space, where distances of a signature representation to the different prototypes are the space constituents. In this space, BFS algorithm is employed to locate the best class specific prototypes. The proposed approach can be employed to develop simple classifiers in the DR space as it results in DRs that are concise, in terms of number of employed feature extractions and prototypes. In addition, designing the DR space through a universal BFS step that is followed by a class-specific BFS step, provides a way to adapt universal classifiers to specific classes. Moreover, the proposed approach can be employed as a feature selection and/or distance function designing tool, where the final classifiers are designed based on the resulting FR space and/or the distance functions. We employed this approach to design classical and bio-cryptographic systems for signature verification through formulating these problems as thresholding classifiers in the resulting DR space. This led to the next two contributions.

The second contribution (Chapter III) focused on the OLSV design problem, where a solution to compromise between pure writer-dependent (WD) and writer-independent (WI) OLSV systems is proposed. The class-independent optimization phase is employed to design WI-SV classifiers, and they are adapted to specific writers through employing a modified version of the class-specific optimization phase. This modification provides designing the final WD classifiers in a reduced FR space, where training is more tractable. Accordingly, no signature templates are stored for verification for enhanced security. It was demonstrated that running the class-specific design phase in the universal FR space produces more accurate representations. This hybrid WI-WD OLSV system enables starting system operation with few signature templates. Switching to a more secure, less complex, and more accurate WD operational mode is possible whenever enough samples are collected for a specific user.

The third contribution (chapter IV) focused on implementing the Fuzzy Vault (FV) scheme based on the offline signature images. The proposed DR optimization approach is employed to

select discriminant features from the signature images that lock/unlock the cryptographic keys within the FVs. Variations of the encoding features are learned in the DR space, and adapt the matching windows during the decoding phase for better recognition accuracy. A simple ensemble of FVs is produced by applying the majority vote decision fusion concept and provided enhanced accuracy. It is important to point out that the proposed FV implementation can be employed as a secure OLSV scheme, besides being used to operate encryption and digital signatures methods based on the handwritten signatures. In the first case, the FV implementation provides template protection, trusted classification decisions, and signature recoverability.

#### **Future research directions:**

The following directions are proposed for future research:

The proposed optimization approach might be applied on other applications (e.g., face recognition, video surveillance, image retrieval, etc). Also, efficiency of the produced DRs should be compared to other distance function learning methods.

In this Thesis class-specific samples are assumed to be collected offline and then they are used for the class-specific optimization phase. Online adaptation of the universal representation might be more practical and facilitates development of adaptive systems. This needs producing an online variant of the BFS algorithm.

Different modules of the DR optimization approach might be enhanced. At the feature extraction level, fusion of more different types of features may enhance the feature representation. For feature selection, feature selection methods other than the BFS algorithm can be investigated.

Adaptation of the class-specific learning phase to the different classes, e.g., applying class-specific signal normalization, training size, etc., might produce more accurate representations.

For instance, for the OLSV application, signature images of the exploitation (class-specific) dataset might need size normalization to match the development dataset signature images.

Concerning the application of the DR optimization approach to the offline signatures, the class-independent learning phase can incorporate simulated forgery samples for training, as these samples might be available in the development database. This way, the resulting universal representation might involve features with higher ability in discriminating simulated forgeries from genuine signatures. In addition, fusion of WI and WD SV modes might enhance the performance during the initial operational period of the OLSV hybrid system.

For the FV implementation, some of the FV parameters may be adapted for each user. For instance, user-specific thresholds can be learned in the DR space, and used to adapt the cryptographic key size. More advanced ensemble methods may be applied to fuse outputs of single FVs in either score or decision levels. Although applied for the offline signatures, the proposed system is generic so can be investigated for other biometrics. Application to multi-modal can also be investigated with either feature, score and decision levels of fusion.

## ANNEX I

### SCENARIOS FOR DISSIMILARITY REPRESENTATION-BASED HANDWRITTEN SIGNATURE VERIFICATION

In this appendix, we explore different scenarios for employing the DR approach for replacing and/or enhancing the standard SV systems. A general framework for designing FR/DR based systems is proposed, that might guide the signature processing research direction to new areas. We argue that the DR approach can be applied in different scenarios, in order to design more robust classifiers. It can enable the design of a new family of classification systems, such as global and hybrid global/user-specific classifiers. Also, the DR approach can be employed, as an intermediate design tool, for enhanced performance of standard feature-based systems. Content of this appendix was published in the International workshop on Automated Forensic Handwriting Analysis (Eskander *et al.*, 2013a).

In Chapter 2, a FR-based DR scheme is introduced, where the DR is build on top of a feature representation (see Figure 2.1). This DR is optimized through applying a two-step learning phases in the dissimilarity space (see Figure 2.2). In this appendix we generalize the DR optimization approach so different tasks for feature selection, prototype selection, and classifiers design, can be done in different spaces, whenever translation between spaces is possible. This strategy permits applying a massive number of pattern recognition techniques, with multiple combinations of space transitions. We propose that new techniques for pattern recognition might be developed based on this strategy. In this context, the DR approach is employed either as a tool for enhancing the standard FR-based systems (for feature/prototype selection), or to design reliable dissimilarity-based classification systems (when classifiers are designed in a DR space).

Figure I-1 illustrates a general framework for designing classification systems based on the DR approach. The standard approach is to extract feature representations from the training

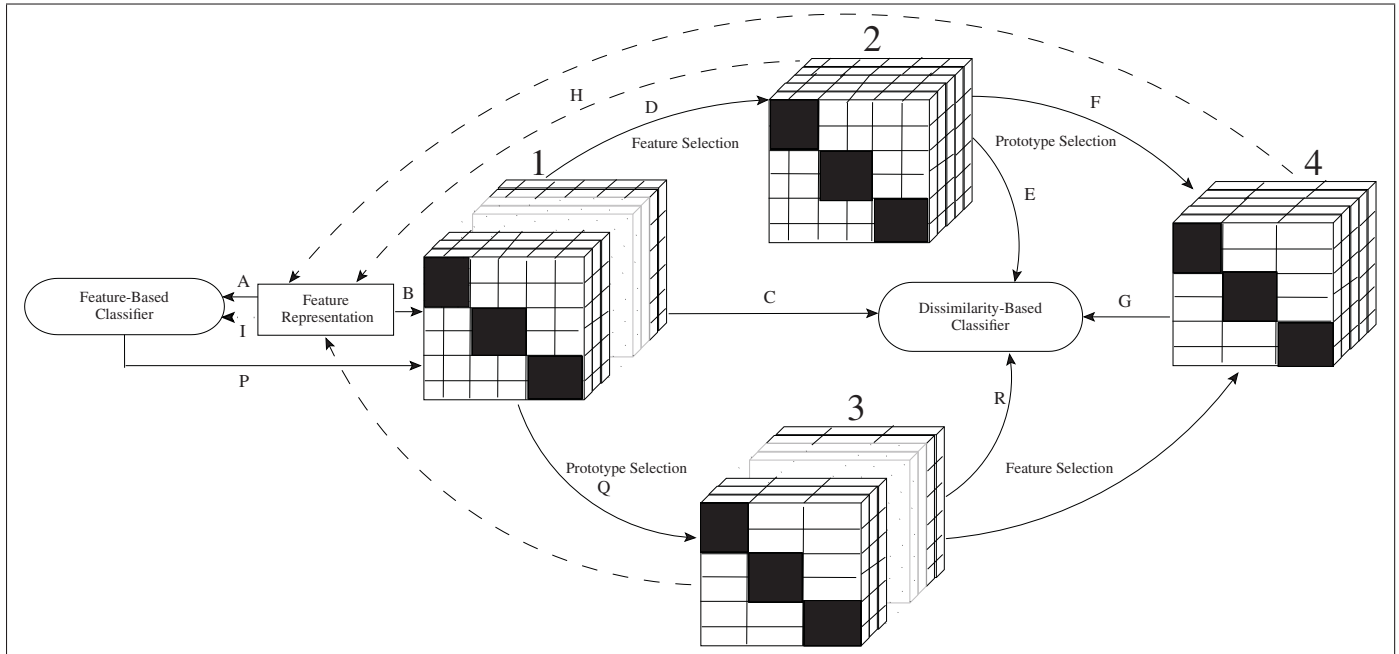


Figure-A I-1 General framework for designing classification systems based on the Dissimilarity Representation (DR) approach: Block 1—full FD-space-full D-space, Block 2—reduced FD-space-full D-space, Block 3—full FD-space-reduced D-space, Block 4—reduced FD-space-reduced D-space.

samples, and design classifiers in the feature space (path A in the Figure). However, the DR approach can be employed in different scenarios for either build new family of classifiers in DR-based spaces, or to enhance the performance of standard feature-based classifiers. More specifically, dissimilarities can be computed on top of a feature representation, and are used to constitute different types dissimilarity representations (DR), e.g., dissimilarity matrix, D-space, or FD-space (path B). The resulting representation could be constituted on top of a huge number of feature extractions, and based on large number of prototypes. The intra-personal (black cells) and inter-personal (white cells) dissimilarities, should be discriminative enough in order to design a DR-based classifier (path C). In case that the DR is not enough informative, feature selection and/or prototype selection can be applied for enhanced representation. For instance, feature selection can be employed in a FD-space (path D). In literature, there are many methodologies of feature selection that can be applied to select the most discriminative and stable features. The resulting DR is constituted on top of a sparser feature representation,

however, redundancy in prototypes may exist (block 2). A classifier can be then designed in the resulting space (path E), or a prototype selection step is done (path F) producing a more compact representation (block 4). Surely, classifiers designed in the sparse and compact representation, are lighter and more accurate (path G). Also, order of the feature/prototype selection processes can be reversed (see the bottom part of the Figure). It is obvious that, it is more logical to run the feature selection process in the FD-space, however, the D-space is more suitable for prototype selection task. The classifier design process can be implemented based on different DR (dissimilarity matrix, D-space, or FD-space).

Besides that the DR approach can be employed to design dissimilarity-based classifiers, it can be considered as an intermediate tool for building reliable feature-based classifiers. Good features and/or prototypes can be selected in a dissimilarity-based space, then the representation is translated back to a sparser and more informative feature space (dotted paths, like path H-I). On contrary, FR-based classifiers can be designed and they are considered as an intermediate tool, to design reliable DR-based classifiers. In such case, multi-classifier systems can be designed, where FR-based classifiers are used to produce the dissimilarity measures, that are needed to build the DR (path P).

## **1. Current Implementations to Offline Signature Systems**

Here we list and categorize some of these implementations, and relate them to the proposed framework for DR-based classification shown in Figure I-1.

### **1.1 Writer-Dependent Systems**

The Writer-Dependent (WD) approach seeks to build a single classifier for each user based on his enrolling signatures. The DR concept is first introduced to design WD-SV systems, by Siteargur N. Srihari et al., Srihari *et al.* (2004). Correlation between high dimensional (1024-bits) binary feature vectors, is employed as a dissimilarity measure. For a specific user,

distances among every pair of his training samples, are determined to represent the intra-class samples. Also, distances between samples of the specific user and some forgeries are computed to represent the inter-class samples. The authors tried different classification strategies: one-class, two-class, discriminative, and generative classifiers. This implementation is a realization of the path B-C in Figure I-1, where classifiers are designed based on the statistics of the dissimilarity matrix.

Later, Batista et al., Batista *et al.* (2010) applied the dissimilarity learning concept to produce reliable WD-SV systems. A feature-based one-class classifier is built by producing user-specific generative models using Hidden Markov models (HMMs). To increase the system accuracy, a two-class discriminative model is build in DR space. The HMMs models are considered as prototypes, and samples are projected to a D-space by considering the likelihood to each HMM generative model as a similarity measure. SVM classifiers are then designed in the produced D-space. This implementation is a realization of the path APC in Figure I-1. Also, the authors employed the AdaBoost method for classifier design in the D-space. This later implementation achieves prototype selection, while building the classifier, which is a realization of the path APQR in the Figure.

## 1.2 Writer-Independent Systems

Instead of building a single writer-dependent (WD) classifier for each user using his enrolling signatures, a single writer-independent (WI) classifier is designed by learning the dissimilarities between signatures of all users. This concept is impossible to realize by means of the standard FR approach. However, it is possible to model the class distributions of intra-class and inter-class dissimilarities, by employing the DR approach. A single "global" classifier can be designed to model, or to discriminate between, these classes. If a huge number of samples are used to build the global DR-based classifier, it is statistically valid that the resulting model generalizes for users whose samples are not included in the training set.



The WI concept is proposed by Siteargur N. Srihari et al., Srihari *et al.* (2004), and Santos and Sabourin et al., C. Santos *et al.* (2004). While the first group used the correlation between binary features as a distance measure, the second group employed the Euclidean distance between graphometric feature vectors. This implementation is a realization of the path BC in Figure I-1, where the classifiers are designed in the FD-space. Improved implementation of this concept is proposed where different dissimilarity spaces are generated based on different feature representations, and classification decisions taken in each space are fused to produce the final decision Oliveira *et al.* (2007), Bertolini *et al.* (2010). This scenario can be considered as generation of different instances for path BC, and fusion is done in the score or decision levels.

More recently, Rivard et al., Rivard *et al.* (2013) extended the idea to perform multiple feature extraction and selection. In this work, information fusion is also performed at the feature level. Multiple graphometric features are extracted based on multiple size grids. Then, the features are fused and pairwise distances between corresponding features are computed to constitute a high dimensional feature-dissimilarity space, where each dimension represents dissimilarity of a single feature. This complex representation is then simplified by applying the boosting feature selection approach (BFS) Tieu and Viola (2004). A sparser and more discriminative FD-space is produced by applying BFS with multi-feature extraction. This scenario can be considered as realization of path BDE in Figure I-1. As the resulting WI classifier recognizes all users, even the users who are enrolled after the design phase, so the feature representation embedded in the WI classifier is considered as a global "population-based" representation.

### 1.3 Adaptation of Writer-Independent Systems

Recently, some work is done to combine advantages of both WI and WD approaches. In Chapter 3, we extend on the system in Rivard *et al.* (2013) by adapting the population-based representation to each specific user, with the aim of reducing the classification complexity. While

the first WI stage is designed in a FD-space, the following WD stage is designed in a standard feature space. Accordingly, the final WD classifier is FR-based classifier, that avoids storing reference signatures for enhanced security. Simulation results on two real-world offline signature databases (the Brazilian DB and GPDS public DB) confirm the feasibility and robustness of the proposed approach. Only a single compact classifier produced similar level of accuracy (Average Error Rate of about 5.38% and 13.96% for the Brazilian and the GPDS databases, respectively) as complex WI and WD systems in literature. This scenario is a realization of path BDHI in Figure I-1.

## **2. Research Directions**

The aforementioned implementations represent a subset of large number of possible FR/DR combinations. Future research may investigate the unvisited scenarios of the proposed framework. For instance, combinations of global/user-specific, generative/discriminative, one-class/two-class systems can be designed. Also, all of the tasks for feature selection, prototype selection, classifier design, etc., can be employed in either feature space, dissimilarity matrix, FD-space, and D-space. Selection of the working space for each step, should depend on the specific requirements and constraints of the design problem and on the application itself. For example, in Chapter 3, features are selected in a FD-space as that provides a way to select reliable feature representations. Then, the classifiers are designed in a standard feature space, to avoid the need for storing signature templates for verification. Besides the large number of possible combinations and translations between the different spaces, there is also a wide range of pattern recognition techniques and tools that can be tested with the proposed framework. This includes different methods for feature extraction and selection, prototype selection, classifiers, etc.

From the application perspective, the proposed framework can be utilized for other applications, rather than the standard SV systems. For example, the Signature Identification (SI)

systems that identify a producer of a signature sample, can be designed based on the DR-approach. Prototypes of all system users can be considered to build a classification D-space. Another example of systems, that imply a challenging design problem, is the signature-based bio-cryptographic systems. In these systems, cryptographic keys of encryption and digital signatures, are secured by means of handwritten signatures. It is a challenging to select informative features, signature prototype, and system parameters, for encoding reliable signature-based bio-cryptographic systems, based on the standard FR approach. In Chapter 4 we discussed a methodology to design such systems, by means of the DR approach. Features are selected in the FD-space and prototypes are selected in the D-space. Some of the system parameters such as length of the cryptographic key, are optimized in the different spaces.

In conclusion, the proposed DR learning framework imparts additional flexibility to the pattern recognition (PR) area. Combinations of transitions between different feature and dissimilarity spaces are suggested. Some of the existing implementations to the SV problem, are surveyed and related to the proposed framework. There are, however, a wide range of methodologies and applications that might benefit from the proposed approach, that opens a door for new research directions.



## ANNEX II

### ADAPTIVE CHAFF GENERATION FOR ENHANCED FV SECURITY

In this appendix, a method to generating chaff points adaptively based on the predicted feature variability is proposed. The method aimed at embedding high number of chaffs for higher entropy against brute-force attack, without impacting much the FV recognition accuracy. Results reported in this appendix was published in the International workshop on Emerging Aspects in Handwritten Signature Processing (Eskander *et al.*, 2013b).

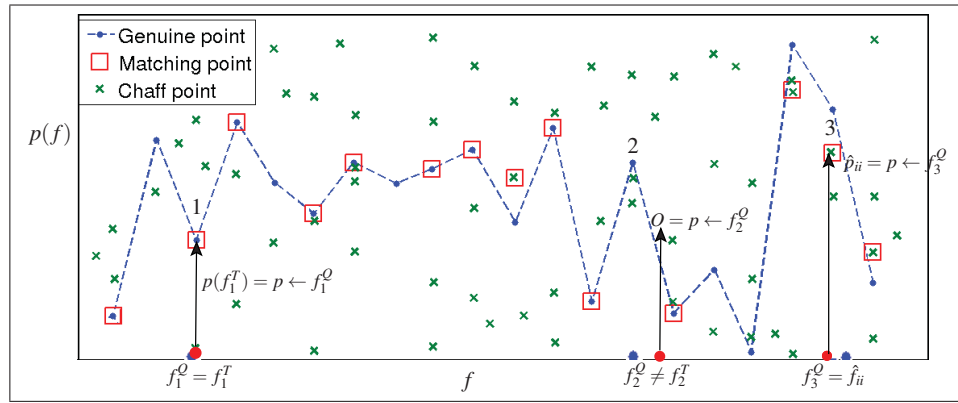


Figure-A II-1 Illustration of the standard locking/unlocking process.

In chapter IV, it is shown that matching FV decoding and encoding points adaptively based on the learned feature variability enhances the FV recognition performance. Here, we apply similar methodology but for generation of the chaff points. Dissimilarity thresholds vector  $\Delta = \{\delta_i\}_{i=1}^t$ , is used during the FV locking phase, for adaptive chaff generation, where  $\delta_i$  is the estimated variability for a feature  $f_i$ . The chaff points are generated so that they have equal separation space  $\Omega$ . This separation space is computed for each feature  $f_i$ , so that  $\Omega_i = 3 \times \delta_i$ . By this method, it is less likely that an unlocking element  $f_i^Q$  equates a chaff element  $f_{ii}$ . For instance, with the tradition FV scheme (see Figure II-1), point  $f_3^Q$  collides with the chaff  $f_{ii}$ , and that results in missing this point in the matching set and that might degrade the FV

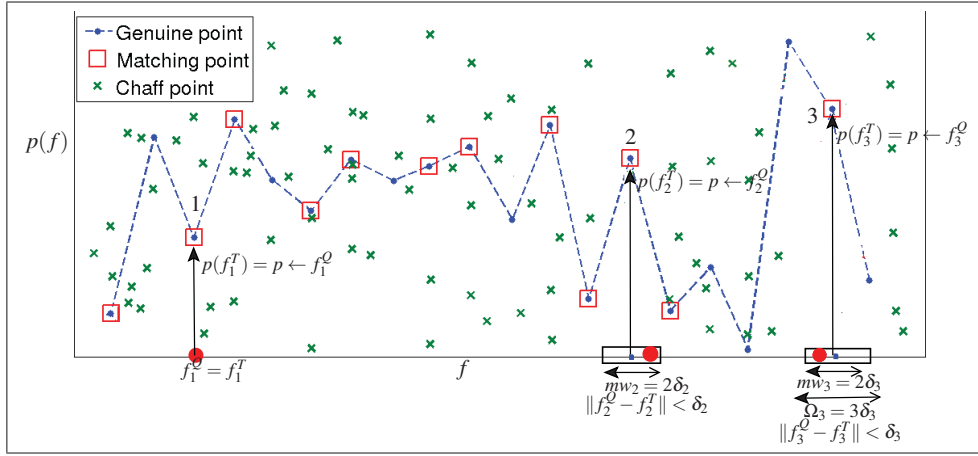


Figure-A II-2 Illustration of proposed adaptive chaff generation method.

recognition performance. On the other hand, with employing the adaptive chaff generation method (see Figure II-2), point  $f_3^Q$  could be filtered and added to the matching set because it did not collide with the chaff  $f_{ii}$ , as the chaff is generated outside the matching window  $w_3$ .

Tableau-A II-1 Impact of Chaff Quantity on the FV Performance

| Chaff separation ( $\Omega$ ) | Without chaff | Fixed separation |         |         |         | Adaptive separation |           |
|-------------------------------|---------------|------------------|---------|---------|---------|---------------------|-----------|
|                               |               | 0.2              | 0.10    | 0.05    | 0.025   | $2\delta$           | $3\delta$ |
| No. of FV points ( $r$ )      | 20            | 200              | 400     | 800     | 1600    | 1768                | 1528      |
| No. of chaff points ( $z$ )   | 0             | 180              | 380     | 780     | 1580    | 1748                | 1508      |
| Security                      | 0-bits        | 45-bits          | 52-bits | 60-bits | 68-bits | 69-bits             | 68-bits   |
| $FRR$                         | 5.25          | 11.53            | 28.94   | 55.53   | 75.81   | 7.03                | 6.13      |
| $FAR_{random}$                | 2.74          | 2.05             | 1.06    | 0.58    | 0.31    | 2.40                | 2.31      |
| $FAR_{simple}$                | 3.49          | 2.39             | 1.58    | 0.88    | 0.49    | 2.89                | 3.26      |
| $FAR_{simulated}$             | 33.14         | 24.38            | 15.63   | 8.15    | 3.42    | 29.77               | 31.06     |
| $AER_{all}$                   | 11.15         | 10.08            | 11.80   | 16.28   | 20.00   | 10.52               | 10.69     |

Table II-1 shows the impact of adaptive chaff generation method. It is clear that the FRR is low when no chaff points are generated, while this implementation is not secure. The traditional chaff generation method, is to generate chaff points with fixed separation space between them. In such case, there is a trade-off between security and robustness. For instance, with small

separation, e.g., 0.025, there are 40 FV points generated with the same index (1 genuine + 39 chaff points). In this case, a high number of chaffs (1580) is generated, while system entropy is 68-bits and  $FRR = 75\%$ . On the other hand, by applying the adaptive chaff generation method, high number of chaffs could be generated (1508), with minimal impact on the system robustness ( $FRR = 6\%$ ).





## **ANNEX III**

### **TOWARDS AUTOMATED TRANSACTIONS BASED ON THE OFFLINE HANDWRITTEN SIGNATURES**

Automating business transactions over the Internet relies on digital signatures, a replacement of conventional handwritten signatures in paper-based processes. Although they guarantee data integrity and authenticity, digital signatures are not as convenient to users as the manuscript ones. In this appendix, a methodology is proposed to produce digital signatures using off-line hand-written signatures. This methodology facilitate the automation of business processes, where users continually employ their handwritten signatures for authentication. Users are isolated from the details related to the generation of digital signatures, yet benefit from enhanced security. First, signature templates from a user are captured and employed to lock his private key in a fuzzy vault. Then, when the user signs a document by hand, his handwritten signature image is employed to unlock his private key. The unlocked key produces a digital signature that is attached to the digitized document. The verification of the digital signature by a recipient implies authenticity of the manuscript signature and integrity of the signed document. Experimental results on the Brazilian off-line signature database (that includes various forgeries) confirms the viability of the proposed approach. Private keys of 1024-bits were unlocked by signature images with Average Error Rate of about 7.8%. The content of this appendix was published in the 9th International conference on Machine Learning and Data Mining (Eskander *et al.*, 2013d).

#### **1. Introduction**

Nowadays, online financial transactions and business agreements are replacing the conventional paper-based processes. One important aspect to accomplish a transaction is to guarantee authenticity of its parties. For the paper-based processes, handwritten signature is the most universally accepted method of authentication. However, identity of the signer is another im-

portant aspect to prove, especially for critical agreements and transactions. Various means are applied to check a signer identity in the paper-based processes. For instance, a signer shows his identity card where a signature is done in front of a legal officer and/or a witness co-signs with the main signer. For the online processes, these conventional methods are not applicable. Instead, the digital signatures can replace the handwritten signatures to authenticate involved parties. The public key infrastructure (PKI) technology is employed for realizing the digital signature concept (Rivest *et al.*, 1978). Two asymmetric keys are generated: a private (signing) key is given for a signer, and a public (verification) key is published to the other parties. To sign a document, the user private key is employed to encrypt some message and attach it to the document. Any party involved in this process can verify the authenticity of the received document. To this end, the recipient extracts the encrypted message from the document, decrypts it by means of the sender's public key, and compares the result with the corresponding plain message. The document is considered authentic, if the two messages are identical. This approach also provides a measure of integrity, as identical messages imply that the document is not tempered while being transferred. On the other hand, integrity of the paper-based processes is hard to proof as document contents can be changed after being signed.

Despite of the enhanced security of the digital signature compared to the handwritten signatures, it has some practical drawbacks. First, digital signatures employ long private keys that is hard to memorize. This problem is alleviated through storing the key in a secure place, e.g., a smart card, and a user retrieves his key by entering a simple password. This scenario, although guarantee authenticity of the digital signature, it does not prove the identity of the signer. Any person, who gets access to the smart card and knows the password, can produce a valid digital signature. Moreover, the security level of the process is degraded, as whatever strong is the signing key, the actual security is determined by the password length. This is known as the cryptographic key management problem. Second, the additional security measures used for digital signatures, like smart cards and passwords, are not as convenient to users as the tradi-

tional manuscript signatures. Moreover, some electronic processes employ paper-based steps that rely on handwritten signatures. For instance, remote bank check deposits imply signing a paper-based check, scan it and submit it remotely to the bank system. Such applications need some integration between the traditional way of authentication and the new technology, which is not offered by the card-password scenario.

In literature, the cryptographic key management problem is alleviated by introducing the bio-cryptography concept (Uludag *et al.*, 2004). Biometrics, that are physical or behavioral human characteristics, are used to control the access to the cryptographic keys. Hence, authenticity of the signer is proved by his traits, instead of something he knows like a password that can be stolen or forgotten. The published bio-cryptographic implementations mostly employed physical biometrics, like fingerprint (Nandakumar *et al.*, 2007), iris (Lee *et al.*, 2008), etc. However, few bio-cryptographic implementations are proposed based on the handwritten signatures. These systems concerned mostly with online systems, where signatures are acquired using special pens and tablets, e.g., (Freire-Santos *et al.*, 2006). These bio-cryptographic implementations, although alleviates the key management problem of digital signatures, they cannot be integrated in applications where the traditional manuscript signatures are employed.

This appendix proposes a digital signature framework based on the offline handwritten signature images. Recently, we introduced a method to secure the cryptographic keys by means of the signature images (Eskander *et al.*, 2011). Here, this method is employed for digital signature key management. To this end, the fuzzy vault (FV) scheme is implemented (Juels and Sudan, 2002), where signature representations are selected through a boosting feature selection (BFS) process (Tieu and Viola, 2004). We show that the proposed method can be employed to manage large keys, e.g., 1024-bits keys, as that involved in the RSA signature schemes (Rivest *et al.*, 1978).

The rest of this appendix is organized as follows. The next section reviews the biometric-based digital signature schemes in the literature. The proposed manuscript signature-based digital signature framework is illustrated in section 3. The experimental methodology is illustrated in section 4. The experimental results are presented and discussed in section 5.

## **2. Biometrics-based digital signatures**

In literature, some of the aforementioned bio-cryptographic implementations are employed to design biometric-based digital signatures. The employed bio-cryptographic schemes are categorized into three main types: 1) key-release, 2) key-generation, and 3) key-binding schemes. In key-release systems, the biometric templates and cryptography keys are stored separately, and the crypto-key is released to genuine users based on classical biometric authentication. To secure both of the key and the template, tamper-resistant storage is needed. In key-generation systems, crypto-keys are generated directly from the biometric traits. This technique is secure, as there is no need to store neither the key nor the biometric template. A drawback of the key-generation approach is that it is hard to generate robust and random keys from unstable and correlating biometric signals. It is also not easy to integrate these biometric-based keys with standard cryptographic algorithms like RSA. Moreover, as private keys are generated directly from the biometric signals, these keys are not revocable (if either the key or the biometric signal is compromised, no new key can be generated). In key-binding systems, classical crypto-keys generated by standard cryptographic keys, e.g., RSA, are coupled with biometric keys. They cannot be decoupled without applying a genuine sample of the biometric trait. Accordingly, reliability and security of key-binding techniques surpasses other cryptography schemes, as they protect the biometric templates and produce typical cryptographic keys.

Janbandhu et al., (Janbandhu and Siyal, 2001) proposed an Iris-based digital signature framework based on 512-bytes Iris templates and a key generation bio-cryptographic scheme. To overcome the irrevocability of the key generation approach, randomly generated numbers are

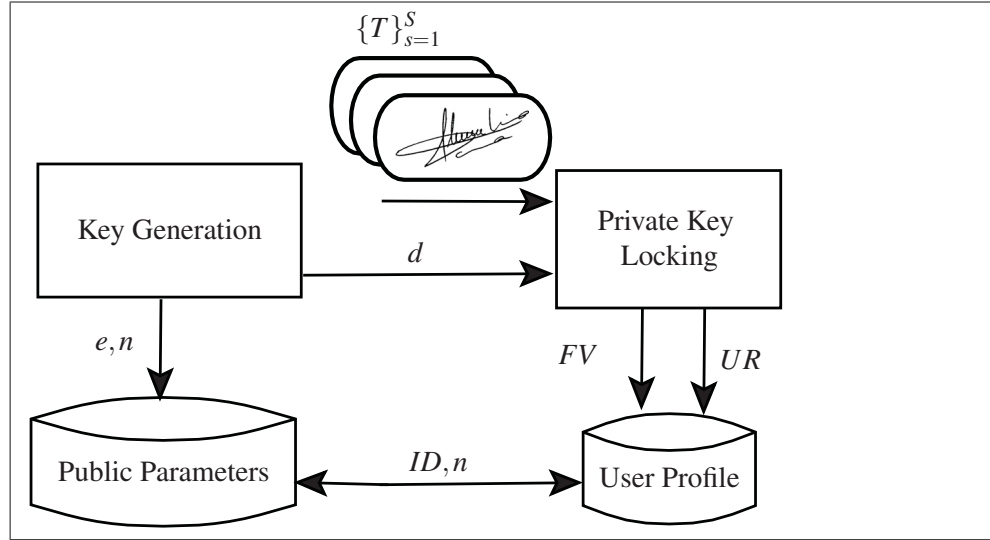


Figure-A III-1 User enrollment process.

employed to modify the iris template. To integrate this scheme with standard public key infrastructures (PKI), e.g., DSA and RSA, the random prime numbers generated by the PKI (to produce the public and private keys), are adjusted to be as close as possible to the Iris template. Mohammadi et al., (Mohammadi and Abedi, 2008) proposed a similar approach, where Iris templates are integrated with elliptic curve cryptography (ECC). Orvos., (Orvos, 2002) introduced a key-binding scheme, where keys of the digital signatures can be encrypted by means of fingerprints. However, this scheme is abstract and no details of the employed key-binding methodology is presented. Kwon (Kwon and Lee, 2004) et al., proposed a fingerprint-based digital signature framework based on a key-binding scheme. The authors employed the concept of key encryption proposed by Soutar et al., (Soutar *et al.*, 1999), where no features are extracted from the fingerprint, but rather an image processing method is applied to lock the private key by the template.

For most of the aforementioned biometric-based digital signature proposals, accuracy of generated keys by means of genuine and forgery biometric signals are not reported. Moreover, the employed biometrics, like iris and fingerprint, might not be user convenient, costly, and not

suitable for some business applications. For instance, it is not practical to accomplish a remote bank check deposit by means of the customer fingerprint, instead signing checks by hand would be more convenient and compatible with the already existing paper-based processes.

### 3. Handwritten signature-based digital signature

#### 3.1 Overview

The proposed system employs the fuzzy vault scheme (FV) (Juels and Sudan, 2002), as a key-binding mechanism for digital signature key management. Typical cryptographic keys are generated through standard public key infrastructures (PKI), and the private key is locked in a secure FV by means of the user handwritten signature image. Later, a user signs his document digitally by providing a genuine handwritten signature sample. Also, a user can delegate a third party for producing digitally signed documents, based on his handwritten signed documents. Any party involved in the transaction, who has the user public key, can validate the digital signature, and hence the authenticity and integrity of the document. The proposed framework consists of three main processes: 1) user enrollment, 2) signing, and 3) verification.

#### 3.2 Enrollment process

Figure III-1 illustrates the user enrollment process. The cryptographic keys are generated according to the employed public key infrastructure (PKI). For instance, for the RSA scheme, a private (signing) key  $K$ , a public key  $e$  and a shared parameter  $q$  are generated for a user. Parameters  $e$  and  $q$  are published to parties, who are supposed to receive and verify documents belonging to the specific user. The private key  $K$  is locked by means of some features, extracted from user's handwritten signature templates  $\{T_s\}_{s=1}^S$ , and constitutes a user fuzzy vault  $FV$ . A user profile contains the FV and user identification data is constituted. This profile might be sent to the user, so he can digitally sign his documents on his own. Also, the user pro-

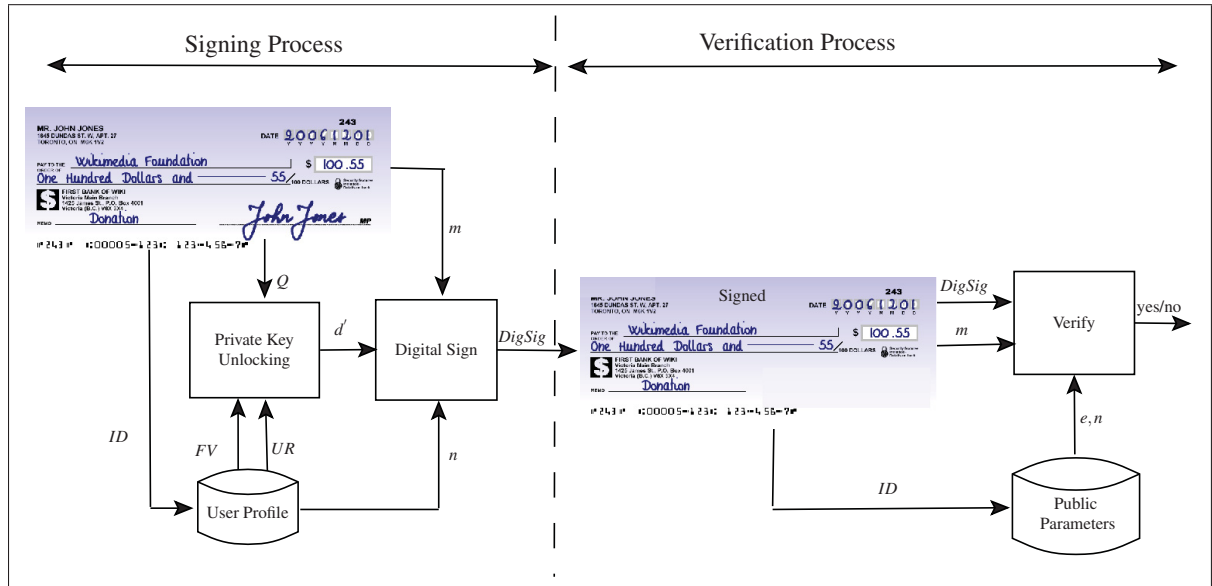


Figure-A III-2 Framework of the proposed digital signature method based on the handwritten signature images.

file might be sent to a trusted party, that can issue digital signatures on behalf of the user. This party extracts the handwritten signatures embedded in the user document, unlocks the user private key  $K$  from the FV by means of the extracted signature image, and then produces a digital signature and attach it to the digitized document. This last scenario simulates the witness party in paper-based agreements or transactions.

### 3.3 Signing process

Figure III-2 (see the left side) illustrates the signing process. To digitally sign a document, the embedded handwritten signature image  $Q$  is extracted and used to decode the user FV. If  $Q$  is genuine, it correctly unlocks the signing key  $K'$  from the FV, where  $K'$  and the original private key  $K$  are identical. The document is then signed by means of  $K'$  and  $q$ . A specific message  $E$  is extracted from the document, e.g., for bank check applications,  $E$  could be the value of the check. Then,  $E$  is encrypted by means of  $K'$  to constitute a digital signature  $DigSig$ , and it is attached to the digitized document.

### 3.4 Verification process

Figure III-2 (see the right side) illustrates the verification process. A recipient, who has the user public key  $e$ , can verify the attached digital signature.  $DigSig$  is decrypted by means of  $e$  and  $q$ , and retrieves the message  $E'$ . The original plain message  $E$  is extracted from the document. The digital signature passes the validation test, if both  $E$  and  $E'$  are identical. In this case, the recipient is sure that the original document is authentic (contains a genuine handwritten signature). Also, this indicates integrity of the document (e.g., the check amount is not changed after the check is signed).

### 3.5 Private key locking and unlocking

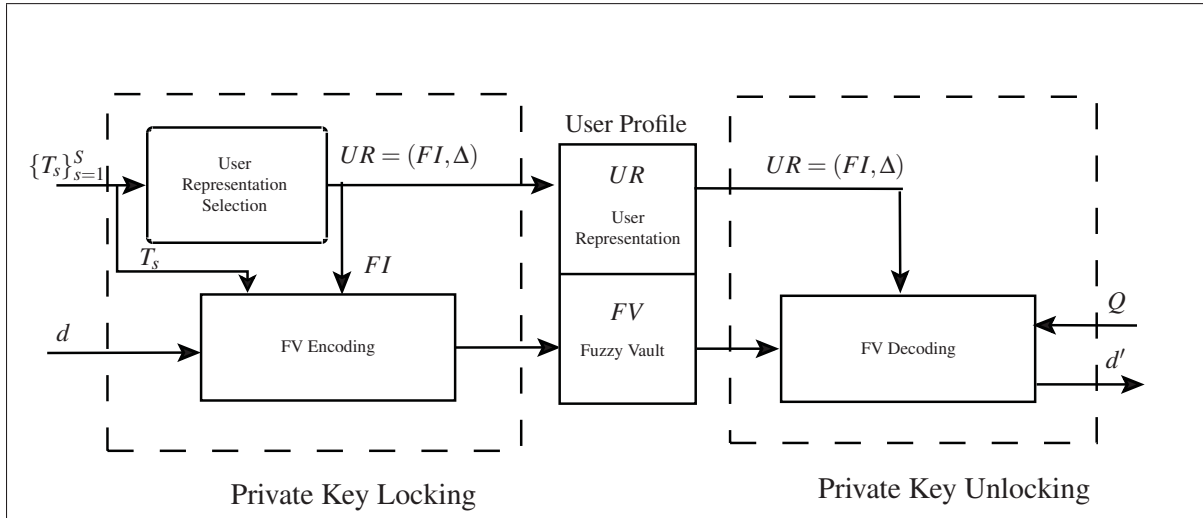


Figure-A III-3 Locking and unlocking of the digital signature private keys within secure FV tokens, by means of the user offline handwritten signature samples.

Figure III-3 illustrates how the private keys are locked and unlocked in/from secure FV tokens. To lock a private key  $K$ , the enrollment handwritten signature templates  $\{T\}_{s=1}^S$  are used to learn a user representation  $UR$ . This representation consists in vectors  $FI$  and  $\Delta$ . Vector  $FI = \{fI_i\}_{i=1}^t$  consists in the indexes of the best  $t$  features, that discriminate between gen-



uine and forgery signatures. Vector  $\Delta = \{\delta_i\}_{i=1}^t$  consists in feature dissimilarity thresholds. Assume  $\delta f_i^{QT}$  is the dissimilarity between two samples  $Q$  and  $T$ , measured by feature  $f_i$ . The dissimilarity feature threshold  $\{\delta_i\}$  is selected so that:  $\delta f_i^{Q_{gen}T} \leq \delta_i$  and  $\delta f_i^{Q_{fr}T} > \delta_i, \forall$  genuine sample  $Q_{gen}$  and forgery sample  $Q_{fr}$ . In a preliminary version of our FV implementation based on the offline handwritten signatures (Eskander *et al.*, 2011), boosting feature selection approach (BFS)(Tieu and Viola, 2004) is employed to select the most discriminative feature  $FI$ . Recently, this feature representation is enhanced by learning the feature dissimilarity vector  $\Delta$  (Eskander *et al.*, 2013f), in a dissimilarity representation space<sup>1</sup>. The features indexes vector  $FI$  is used to extract a feature representation  $F^T = \{F_i\}_{i=1}^t$  from a prototype signature  $T_s$ . Then,  $F^T$  locks the private key  $K$  in a FV. To this end,  $K$  is used to generate an encoding polynomial  $p$ , by splitting  $K$  of size  $KS$ -bits into  $k + 1$  equal chunks ( $c$ ) of size  $l$ -bits. These chunks are used as polynomial coefficients. So that the encoding polynomial  $p$  is given by:

$$p(a_i) = c_k a_k^k + c_{k-1} a_{k-1}^{k-1} + \dots + c_1 a + c_0. \quad (\text{A III-1})$$

The extracted features  $F^T$  are quantized in elements of  $l$ -bit, to constitute a locking vector  $A = \{a_i\}_{i=1}^t$ . Genuine FV points  $\{A, P(A)\}$  are constituted by computing the polynomial, given by Eq.A III-1, for all elements of  $A$ . To hide the genuine points,  $z$  chaff (noise) points  $\{\hat{A}, \hat{P}\}$  are generated, so that they do not collide with the genuine points. Finally both genuine and chaff points are merged to constitute  $r$  FV points  $\{\tilde{A}, \tilde{P}\}$ , where  $r = t + z$ .

To unlock the private key  $K$ , a query handwritten signature image  $Q$  is extracted from the document, and used to decode the FV. A query feature vector  $F^Q$  is extracted from  $Q$ , based on the pre-selected feature indexes  $FI$ . Feature quantization is done, as that for the FV encoding process, and the FV unlocking vector  $B = \{b_i\}_{i=1}^t$  is constituted. Elements of  $B$  are matched

---

<sup>1</sup>Details of the BFS and dissimilarity learning is out of the scope of this appendix. For more details, see (Eskander *et al.*, 2011) and (Eskander *et al.*, 2013f). This method relies on recent works proposed by Rivard *et al.*, (Rivard *et al.*, 2013) and Eskander *et al.*, (Eskander *et al.*, 2012), for designing writer-independent and writer-dependent offline signature verification systems, respectively.

against all points of the FV, so that the chaff points are filtered out. For the preliminary version of our FV implementation (Eskander *et al.*, 2011), elements of  $A$  and  $B$  are strictly matched. Two elements are considered matching if they have exact quantized values. In this work, the modeled dissimilarity threshold vector  $\Delta$  is used for adaptively matching elements based on their expected dissimilarities (Eskander *et al.*, 2014a). Two elements are considered matching if their dissimilarity is less than the corresponding dissimilarity threshold. The resulting vector  $\{\bar{A}, \bar{P}\}$  is used to reconstruct the polynomial  $p'$ , by applying the Reed-Solomon error correction codes (Berlekamp and Elwyn, 1968). Finally, the coefficients of  $p'$  are assembled to constitute the key  $K'$ . If the FV is correctly decoded, the unlocked key  $K'$  is identical to the user private key  $K$ . For more details about the FV encoding and decoding processes and the dissimilarity representation, see (Eskander *et al.*, 2011) and (Eskander *et al.*, 2014a).

#### 4. Experimental methodology

The Brazilian database (Freitas *et al.*, 2000) is used for proof-of-concept simulations. This DB contains three types of signature forgery: random, simple and simulated. Random forgeries do not know neither the writer's name nor the signature morphology. For simple forgery, the forger knows the writer's name and he produces a simple forgery using his writing style. A simulated forgery has access to a sample of the signature and imitates the genuine signature. The signatures were provided by 168 writers. Signatures of first 60 writers include: 40 genuine signatures, 10 simple forgeries and 10 simulated forgeries per writer. Of them, 30 genuine signatures, besides some of signatures selected from the last 108 users (that represents random forgeries), are used for the user representation ( $UR$ ) selection task. For performance evaluation, the rest 10 genuine samples, 10 simple, 10 simulated forgeries, and 10 random forgeries (belong to the last 108 users) are used.

In the preliminary version of this work (Eskander *et al.*, 2011), we employed extended-shadow-code (ESC) features (Sabourin and Genest, 1994). Here, we investigate a multi-type feature

extraction approach, where directional probability density function (DPDF) (Drouhard *et al.*, 1996) is also employed. Features are extracted based on 30 different grid scales producing 60 different single scale feature representations. These representations are then fused to produce a feature representation of huge dimensionality (30, 201) (Rivard *et al.*, 2013).

For digital signature key generation, the RSA scheme is employed (Rivest *et al.*, 1978). Keys of different sizes are generated, where the private key  $K$  is locked in a FV. Digital signatures is produced by means the private key  $K'$ , unlocked from the FV by means of query signatures. Verification of the digital signatures is done by means the public key  $e$ .

The FV parameters are set as follows: the quantization size  $l$  is set to 16-bits. Different key sizes ( $KS$ ) are employed (128, 256, 512, 1024-bits). Number of genuine FV locking points  $t$  is determined experimentally for the different key sizes.

The Average Error Rate ( $AER$ ) is employed for performance evaluation and computed as follows:

$$AER = (FRR + FAR_{random} + FAR_{simple} + FAR_{simulated})/4. \quad (\text{A III-2})$$

False Reject Rate ( $FRR$ ) is the ratio of genuine query signatures that failed to decode the FV, and produce valid digital signatures.  $FAR_{random}$ ,  $FAR_{simple}$  and  $FAR_{simulated}$  are the ratio of random, simple, and simulated forgeries, respectively, that succeed to decode the FV, and produce valid digital signatures.

## 5. Experimental results

Table III-1 reports the experimental results for different key sizes. In (Eskander *et al.*, 2011), only ESC features are employed and features are matched strictly. Here, when the DPDF features are added and features are matched adaptively, the performance is enhanced as  $AER$  is reduced from 17.75% to 10.08%. It is found that the proposed FV method could secure large keys with acceptable accuracy, so it could be integrated in practical digital signature schemes

Tableau-A III-1 Performance of the proposed manuscript signature-based digital signatures.

| Parameters | $KS$ -bits        | 128 (previous work)<br>(Eskander <i>et al.</i> , 2011) | 128   | 256   | 512   | 1024  |
|------------|-------------------|--|-------|-------|-------|-------|
|            | $t$               | 20   | 20    | 40    | 140   | 200   |
|            | $z$               | 180  | 180   | 360   | 1260  | 1800  |
| Measure %  | $FRR$             | 25   | 11.53 | 13.55 | 5.31  | 11.26 |
|            | $FAR_{random}$    | 3  | 2.05  | 2.00  | 2.71  | 0.98  |
|            | $FAR_{simple}$    | 7  | 2.39  | 2.28  | 3.31  | 1.31  |
|            | $FAR_{simulated}$ | 36   | 24.38 | 19.28 | 29.26 | 17.43 |
|            | $AER$             | 17.57  | 10.08 | 9.27  | 10.14 | 7.75  |

like RSA. However, the size of the FV locking vector  $t$  should be increased for the large keys. Also, two important aspects are noticed: 1) different key sizes result in different performance for the different users. This motivate future investigation for adapting the key length for each user, 2) the performance differs for the different signature templates that locks the FV. This motivates further work to address prototype selection for FV encoding (Eskander *et al.*, 2013f).

## 6. Conclusions and future work

A framework for digital signature by means of the handwritten signature images is proposed. The private keys are locked by the user signature templates and released only when a user provides a genuine signature sample. This framework facilitates various industrial applications like remote bank check transactions, ATM and credit card transactions, automation of business procedures including legal agreements, etc. Also, the proposed system permits delegation of authority, as a third party who stores the user profile, can generate signed digitized documents on behalf of a user based on his embedded handwritten signature. A fuzzy vault system is designed to protect the signature keys, where boosting feature selection process is employed in a dissimilarity representation space. Accuracy of FV decoding is increased through applying multi-type feature extraction and matching the FV encoding and decoding features adaptively, based on the modeled feature dissimilarity. Decoding accuracy of genuine, random and

simple forgeries is acceptable, even with large cryptography keys. Resistance of the system against simulated forgeries needs enhancement, and further work is needed to detect this type of forgery. Also, future work should be conducted to increase the system accuracy, through adapting the FV parameters and select the signature prototypes for the specific users (Eskander *et al.*, 2013f).



## ANNEX IV

### **IMPROVING SIGNATURE-BASED BIOMETRIC CRYPTOSYSTEMS USING CASCADED SIGNATURE VERIFICATION-FUZZY VAULT (SV-FV) APPROACH**

Signature-based FVs are designed in a dissimilarity space, where distances between feature pairs are the space constituents. Feature representations selected in such dissimilarity spaces have shown acceptable level of robustness against signature variability, while they lack discriminative power against skilled forgeries Eskander *et al.* (2014a).

In chapter 4, the limited discriminative power of FVs is alleviated by using an additional password, so that the false accept rate (FAR) is reduced without significantly affecting the false reject rate (FRR). However, enhancing system accuracy comes with the expense of the user inconvenience.

In this appendix, a novel user-convenient approach is proposed for enhancing the accuracy of signature-based biometric cryptosystems. Since signature verification (SV) systems designed in the original feature space have demonstrated higher discriminative power to detect impostors Eskander *et al.* (2013c), they can be used to improve the FV systems. Instead of using an additional password, the same signature sample is processed by a SV classifier before triggers the FV decoders. Using this cascaded approach, the high FAR of FV decoders is alleviated by the higher capacity of SV classifiers to detect impostors. The content of this appendix is published in the 14th International conference on Frontiers in Handwriting Recognition (ICFHR-2014) (Eskander *et al.*, 2014b).

To this end, SV module is designed in the feature space by applying a two-step BFS approach as proposed in Chapter 3. Also, FVs are produced through a dissimilarity based approach as illustrated in Chapter 4. During authentication, a query signature is verified by the SV module. The sample is processed by FV decoders for cryptographic key unlocking, only if it is verified by SV module.

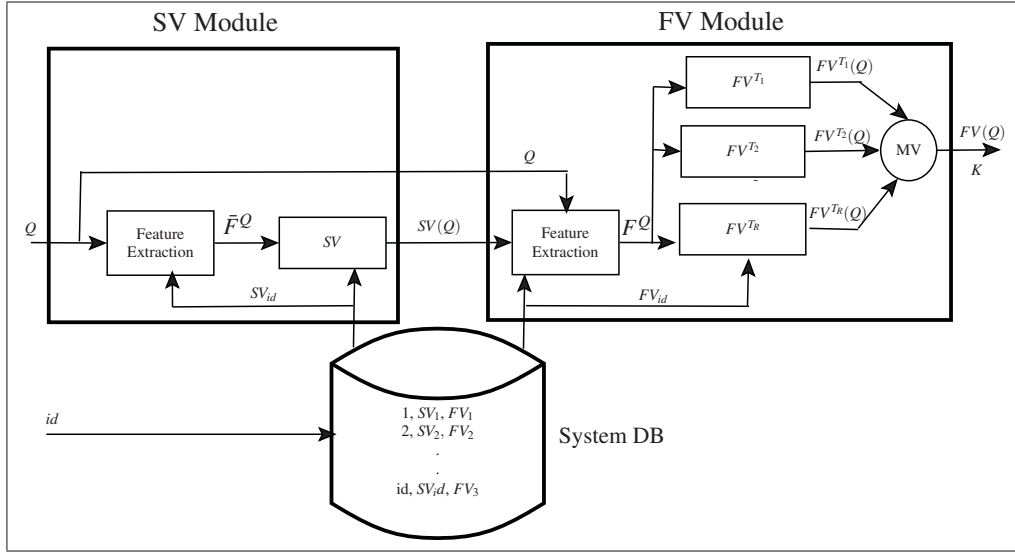


Figure-A IV-1 Proposed cascaded SV-FV system in the verification mode: different feature representations  $\bar{F}$  and  $F$  are processed by a SV classifier and a set of FV decoders, respectively. The FV module is triggered only if the SV module produces a positive classification label.

Figure IV-1 illustrates the proposed cascaded SV-FV system in the verification mode. First, a feature representation  $\bar{F}^Q$  is extracted from a query signature  $Q$  according to the WD-SV representation stored in system database. The WD-SV classifier is used to verify the signature sample as illustrated in Chapter 3. Since SV classifiers have demonstrated higher discriminative power than that of the FV systems, so a range of forgeries are filtered at this step.

Then, the FV module is triggered only if  $Q$  is verified as a genuine signature, ie.,  $SV(Q) = 1$ . In such case, a feature representation  $F^Q$  is extracted from  $Q$  according to the user FV stored in system database. Instead of unlocking a single FV, a set of FV templates  $\{FV^{T_r}\}_{r=1}^R$  are used for improved recognition. Every FV is unlocked as illustrated in Chapter 4, where a FV produces a positive label ( $FV(Q) = 1$ ) when it is successfully unlocked. Finally, a majority vote ( $MV$ ) rule is applied so that unlocked cryptographic key  $K$  is released to the user only if majority of FVs are successfully unlocked.



The Brazilian database Freitas *et al.* (2000) is used for proof-of-concept simulations. To design the FV module, the first 30 genuine signatures of each user are used to produce 30 different FVs/user. For FV encoding,  $t = 20$  genuine vault points are produced from a signature image. To this end, the feature values are quantized in  $l = 8$ -bits. To conceal the genuine points,  $z = 180$  chaff points are injected, so total number of FV points  $r = 200$ . Length of encoded cryptographic key  $K$  is 128-bits, that encodes a polynomial  $p$  of degree  $k = 7$ .

To design the SV module, features are extracted from the 30 genuine signature/user. A single SV classifier is produced for each user, where a BFS process runs for 100 boosting iterations. Average dimensionality of the resulting WD representations is ( $N = 40$ ). A zero classification threshold is used in all experiments ( $\theta = 0$ ).

To investigate the viability of the proposed cascading approach, both pure FV and SV systems are compared with the cascaded SV-FV system. The impact of fusion of multiple FVs on the decision level is tested by repeating the experiments for single FV decoding ( $R = 1$ ) and for multiple FVs where  $R = 15$ .

For the 60 users in the testing set, 40 test samples per user are employed (10 genuine, 10 random, 10 simple, and 10 skilled forgeries). In case of FV systems (either pure FVs or cascaded SV-FV systems), each test sample is verified against the 30 genuine FV templates, for a total of 72000 ( $60 \times 40 \times 30$ ) verification trials. For the SV systems, each of the 40 queries per user are verified once against the SV model, for a total of 2400 ( $60 \times 40$ ) verification trials. For performance evaluation, the average error rate ( $AER$ ) is computed as follows:

$$AER = (FRR + FAR_{rand} + FAR_{smp} + FAR_{skl})/4. \quad (\text{A IV-1})$$

where  $FAR_{rand}$ ,  $FAR_{smp}$ , and  $FAR_{skl}$  are computed for random, simple, and skilled forgeries, respectively.

Tableau-A IV-1 Comparison of the SV and FV systems to the proposed cascaded SV-FV system

| Performance measure | Method  |         |          |                   |          |
|---------------------|---------|---------|----------|-------------------|----------|
|                     | SV only | FV only |          | Proposed approach |          |
|                     |         | $R = 1$ | $R = 15$ | $R = 1$           | $R = 15$ |
| $FRR$               | 7.83    | 11.53   | 7.69     | 14.00             | 10.87    |
| $FAR_{random}$      | 0.01    | 2.05    | 5.02-05  | 0.00              | 0.00     |
| $FAR_{simple}$      | 0.17    | 2.39    | 1.92     | 0.30              | 0.33     |
| $FAR_{skilled}$     | 13.50   | 24.38   | 23.26    | 10.80             | 11.38    |
| $AER$               | 5.38    | 10.08   | 8.21     | 6.55              | 5.64     |

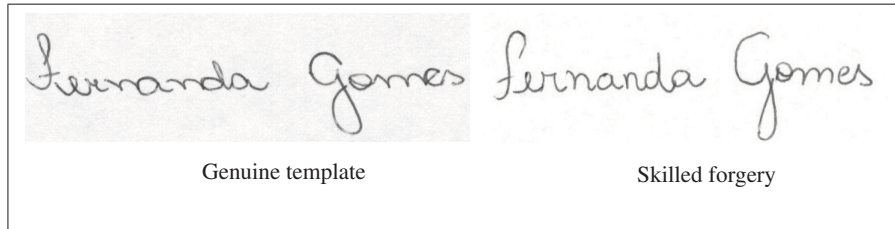


Figure-A IV-2 Example of skilled forgeries that are filtered by the SV module.

Performance of dissimilarity-based FVs is encouraging. For instance, for a user with signatures shown in Figure 4.4, the single-type–single-resolution feature extraction technique produced FVs with  $FRR = 100\%$ . In this case, out of the 20 locking points, only 10 points are filtered by the unlocking sets (error = 10). For FVs with error correction capability  $\epsilon = 6$ , these errors are not canceled by the FV decoder. On the other hand, applying multi-feature extraction and the dissimilarity-based BFS approach produced FVs with  $AER = 0\%$ , as the mismatch errors are within the FV error correction capacity (error < 6).

However, due to FV scheme limitations, such accurate recognition does not apply for all users in the testing database. For instance, for the user with signatures shown in Figure IV-2, although his signatures are stable ( $FRR = 0\%$ ), they are easy to imitated by skilled forgeries. For this user,  $FAR_{skilled} = 16\%$ .

On the other hand, SV classifiers have shown higher accuracy due to the relatively complex model of the SV classifiers, as compared to simple FVs. For instance, for the user with signatures shown in Figure IV-2, all skilled forgeries are detected by the SV classifier.

In case of the proposed cascaded SV-FV solution, the accuracy is enhanced through filtering most of forgeries by the SV classifier. For instance, impostor signature, shown in Figure IV-2 (right), could unlock a FV by the template shown in Figure IV-2 (left), when a pure FV is used. On the other hand, for the cascaded SV-FV solution, this forgery is detected by the SV classifier and it is filtered before triggering the FV. For this user,  $FAER_{skilled} = 0\%$ , when the cascaded SV-FV system is employed.

Table IV-1 presents the simulation results for all users in the testing dataset. The pure FV system, with a single template ( $R = 1$ ), has shown  $AER = 10.08\%$ . When multiple FVs are decoded ( $R = 15$ ), the AER is reduced by 18.5% (from 10.08% to 8.21%). However, this comes with expense of increased decoding complexity. For the pure SV systems, the performance is much better ( $AER = 5.38\%$ ). However, this solution produces simple classification labels and can not secure cryptographic keys.

In case of the cascaded SV-FV solution, the AER of cryptographic key decoding is decreased by 35% (from 10.08% to 6.55%). More specifically, accuracy of detecting skilled forgeries is much increased, with out high impact on the genuine accept rate, where  $AER_{skilled}$  is decreased by 58.65% (from 24.38% to 10.80%). When multiple FVs are fused ( $R = 15$ ), the  $AER$  is decreased by 13.89% (from 6.55% to 5.64%). Generally, this result is comparable to that of the pure SV classifier. Hence, using the proposed approach facilitates securing cryptographic keys by means of offline signature images with similar level of accuracy as that of the classical SV systems.



## BIBLIOGRAPHY

- A. Menezes, P. van Oorschot, S. Vanstone, 1996. *Hand book of applied cryptography*. CRC press.
- Anil K. Jain, Arun Ross, Salil Prabhakar. 2004. "An Introduction to Biometric Recognition". *IEEE transactions on circuits and systems for video technology*, vol. 14, p. 4-20.
- Babenko, B., S. Branson, and S. Belongie. 2009. "Similarity Metrics for Categorization: from Monolithic to Category Specific". In *Int'l Conf. on Computer Vision*. (Kyoto, Japan 2009), p. 293-300.
- Bar-Hillel, A., T. Hertz, N. Shental, and D. Weinshall. 2005. "Learning and Mahalanobis Metric from Equivalence Constraints". *Machine Learning Research*, vol. 6, p. 937-965.
- Batista, L., D. Rivard, R. Sabourin, E. Granger, and P. Maupin, 2007. *State of the art in off-line signature verification*. IGI Global.
- Batista, L., E. Granger, and R. Sabourin. 2010. "Applying Dissimilarity Representation to Off-Line Signature Verification". In *Int'l Conf. on Pattern Recognition*. p. 1293-1297.
- Batista, L., E. Granger, and R. Sabourin. 2012. "Dynamic Selection of Generative-Discriminative Ensembles for Off-Line Signature Verification". *Pattern Recognition*, vol. 45, n° 4, p. 1326-1340.
- Bengio, S. and J. Marithoz. 2007. "Biometric Person Authentication IS A Multiple Classifier Problem". *7th Int'l Workshop on Multiple Classifier Systems, LNCS*, vol. 4472, p. 513-522.
- Berlekamp and R. Elwyn, 1968. *Algebraic Coding Theory*. McGraw-Hill.
- Bertolini, D., L. Oliveira, E. Justino, and R. Sabourin. 2010. "Reducing forgeries in Writer-Independent Off-line Signature Verification Through Ensemble of Classifiers". *Pattern Recognition*, vol. 43, n° 1, p. 387-396.
- Breiman, J. H., R. A. Friedman, Olshen, and C. J. Stone. 1984. "Classification and Regression Trees". *Wadsworth and Brooks*.
- Bunke, M. Lastand H. and A. Kandel. 2002. "A feature-based serial approach to classifier combination". *Pattern Analysis and Applications*, p. 385-398.
- Burr, W. E., D. F. Dodson, and W. T. Polk, 2006. *Information security: electronic authentication guideline*. NIST Special Publication.
- C. Santos, E. Justino, F. Bortolozzi, and R. Sabourin. 2004. "An Off-line Signature Verification Method Based on Document Questioned Experts Approach and a Neural Network Classifier". In *Proc. of 9th Int'l Workshop on Frontiers in Handwriting Recognition*. (Tokyo, Japan 2004), p. 498-502.

- Cha, S. 2001. "Use of distance measures in handwriting Analysis". PhD thesis, State University of New York at Buffalo, USA.
- Cha, Sung-Hyuk. 2007. "Comprehensive Survey on Distance/Similarity Measures between Probability Density Functions". *Int'l Journal of Mathematical Models and Methods in Applied Sciences*, vol. 1, n° 4, p. 300-307.
- Clancy, T. Charles, Negar Kiyavash, and Dennis J. Lin. 2003. "Secure smartcard-based fingerprint authentication". In *ACM Workshop on Biometric Methods and Applications*. (Berkley, CA, USA 2003), p. 45-52.
- Domeniconi, C., J. Peng, and D. Gunopulos. 2002. "Locally Adaptive Metric Nearest-neighbor Classification". *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 24, n° 9, p. 1281-1285.
- Drouhard, J., R. Sabourin, and M. Godbout. 1996. "A Neural Network Approach to Off-line Signature Verification Using Directional PDF". *Pattern Recognition*, vol. 29, n° 3, p. 415-424.
- Duin, Robert P.W., Marco Loog, Elzbieta Pekalska, and David M.J. Tax. 2010. "Feature-Based Dissimilarity Space Classification". In *20th Int'l conf. on Recognizing Patterns in Signals, Speech, Images, and Videos*. p. 46-55.
- Eskander, G., R. Sabourin, and E. Granger. 2011. "Signature based Fuzzy Vaults with boosted feature selection". In *IEEE Workshop on Computational Intelligence and Identity Management*. (Paris 2011), p. 131-138.
- Eskander, G., R. Sabourin, and E. Granger. 2012. "Adaptation of writer-independent systems for offline signature verification". In *The 13th Int'l Conference on Frontiers in Handwriting Recognition*. (Bari, Italy, 2012), p. 432-437.
- Eskander, G., R. Sabourin, and E. Granger. 2013a. "Dissimilarity Representation for Handwritten Signature Verification". In *2nd Int'l Workshop on Automated Forensic Handwriting Analysis*. (Washington DC, USA 2013), p. 371-376.
- Eskander, G., R. Sabourin, and E. Granger. 2013b. "A Dissimilarity-Based Approach for Biometric Fuzzy Vaults—Application to Handwritten Signature Images". In *Int'l Workshop on Emerging Aspects in Handwritten Signature Processing*. (Naples, Italy 2013).
- Eskander, G., R. Sabourin, and E. Granger. 2013c. "Hybrid Writer-Independent–Writer-Dependent Offline Signature Verification System". *IET-Biometrics Journal, Special issue on Handwriting Biometrics*, vol. 2, n° 4, p. 169-181.
- Eskander, G., R. Sabourin, and E. Granger. 2013d. "Towards Automated Transactions based on the Offline Handwritten Signatures". In *9th Int'l Conference on Machine Learning and Data Mining*. (New York, USA 2013).

- Eskander, G., R. Sabourin, and E. Granger. 2013e. "Optimized Dissimilarity Representations With Application to Signature Verification and Bio- Cryptography". *Special issue of the IEEE Transactions on Neural Networks and Learning Systems on Learning in non-(geo)metric spaces (submitted)*.
- Eskander, G., R. Sabourin, and E. Granger. 2014a. "Bio-Cryptographic System Based on Offline Signature Images". *Information Sciences*, vol. 259 (2014), p. 170–191.
- Eskander, G., R. Sabourin, and E. Granger. 2014b. "Improving Signature-Based Biometric Cryptosystems Using Cascaded Signature Verification–Fuzzy Vault (SV-FV) Approach". In *The 14th Int'l Conf. on Frontiers in Handwriting Recognition*. (Crete Island, Greece 2014).
- Eskander, G.S., R. Sabourin, and E. Granger. 2013f. "On the Dissimilarity Representation and Prototype Selection for Signature-Based Bio-Cryptographic Systems.". In *2nd Int'l. Workshop on Similarity-Based Pattern Analysis and Recognition*. (York, UK 2013), p. pp.265-280.
- Fisher, R. A. 1936. "Use of Multiple Measurements in Taxonomic Problems". *Annals of Eugenics*, vol. 7, n° 2, p. 179-188.
- Franssen, T., H. Darmstadt, D.X. Zhou, and C Busch. 2008. "Fuzzy Vault for 3D face recognition systems". In *Int'l Conference on Intelligent Information Hiding and Multimedia Signal Processing*. (Harbin, China 2008), p. 1069-1074.
- Freire, Manuel R., Julian Fierrez, and Javier Ortega-garcia. 2008. "Dynamic signature verification with template protection using helper data". In *IEEE Int'l Conference on Acoustics, Speech and Signal Processing*. (Las Vegas, Nevada, USA 2008), p. 1713-1716.
- Freire-Santos, M., J. Fierrez-Aguilar, and J. Ortega-Garcia. 2006. "Cryptographic key generation using handwritten signatures". In *proc of SPIE*. p. 225-231.
- Freire-Santos, Manuel, J. Fierrez-Aguilar, M. Martinez-Diaz, and J. Ortega-Garcia. 2007. "On the applicability of off-line signatures to the Fuzzy Vault construction". In *ICDAR2007*. (Curitiba, Brazil 2007).
- Freitas, C., M. Morita, L. Oliveira, A. Yacoubi E. Justino, E. Lethelier, F. Bortolozzi, and R. Sabourin. 2000. "Bases de dados de cheques bancarios brasileiros". In *XXVI Conferencia Latinoamericana de Informatica*. (México 2000).
- Frome, A., Y. Singer, and J. Malik. 2007. "Image Retrieval and Classification Using Local Distance Functions". *Advances in Neural Information Processing Systems*, vol. 19, p. 417-424.
- Garcia, Salvador, Joaquin Derrac, Jose Ramon Cano, and Francisco Herrera. 2012. "Prototype Selection for Nearest Neighbor Classification: Taxonomy and Empirical Study". *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 34, n° 3, p. 417-435.



- Guyon, Isabelle and Andre Elisseeff. 2003. "An Introduction to Variable and Feature Selection". *Journal of Machine Learning Research*, vol. 3, p. 1157-1182.
- Hirschbichler, M. 2008. "A Multiple-Control Fuzzy Vault". In *Sixth Annual Conference on Privacy, Security and Trust*.
- Hoque, S, M Fairhurst, and G Howells. 2008. "Evaluating biometric encryption key generation using handwritten signatures". In *ECSIS Symposium on Bio-inspired Learning and Intelligent Systems for Security*. (Edinburgh, UK 2008), p. 17-22.
- Iba, W. and P. Langley. 1992. "Induction of One-level Decision Trees". In *Proc. of the 9th Int'l Machine Learning Conf.* (Scotland 1992), p. 233-240.
- Impedovo, D. and G. Pirlo. 2008. "Automatic Signature Verification: the State of the Art". *IEEE Trans. on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 38, n° 5, p. 609-635.
- Jain, A. K., L. Hong, and R. Bolle. 1997. "On-line fingerprint verification". *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, n° 4, p. 302-3147.
- Jain, Anil. K., Arun Ross, and Sharath Pankanti. 2006. "Biometrics: A Tool for Information Security". *IEEE Trans. on Information Forensics and Security*, vol. 1, n° 2, p. 125-143.
- Janbandhu, P and M. Siyal. 2001. "Novel biometric digital signatures for Internet-based applications". *Information Management and Computer Security*, vol. 9, n° 5, p. 205-21.
- Juels, A. and M. Sudan. 2002. "A Fuzzy Vault Scheme". In *Proc. IEEE Int'l Symp. of Information Theory*. p. 408.
- Juels, A. and M. Wattenberg. 1999. "A Fuzzy Commitment Scheme". In *Proc. of the 6th ACM Conf. on Computer and Communications Security*. p. 28-36.
- Justino, E., F. Bortolozzi, and R. Sabourin. 2001. "Off-line signature verification using HMM for random, simple and skilled forgeries". In *Sixth Int'l Conference on Document Analysis and Recognition*. (Seattle (Washington, USA) 2001), p. 1031-1034.
- Khreich, W., E. Granger, A. Miri, and R. Sabourin. 2010. "Iterative Boolean combination of classifiers in the ROC space: An application to anomaly detection with HMMs". *Pattern Recognition*, vol. 31, p. 2732-2752.
- K.Nandakumar, A. Nagar, and A. K. Jain. 2007. "Hardening fingerprint Fuzzy Vault using password". *Proc. Int' conference on Advances in Biometrics, LNCS*, vol. 4642, p. 927-937.
- Kohavi and G. John. 1997. "Wrappers for Feature Selection". *Artificial Intelligence*, vol. 97, p. 273-324.
- Kumar, Amioy and Ajay Kumar. 2009. "Development of a new cryptographic construct using Palmprint based Fuzzy Vault". *EURASIP Journal on Advances in Signal Processing*, vol. 2009, n° 13.



- Kumar, R., J. Sharma, and B. Chanda. 2012. "Writer-independent off-line signature verification using surroundedness feature". *Pattern Recognition Letters*, vol. 33, n° 3, p. 301-308.
- Kwon, T and J Lee. 2004. "Practical digital signature generation using biometrics". *Lecture Notes in Computer Science*, vol. 3043, p. 728-737.
- Lee, Y. J., K. R. Park, S. J. Lee, K. Bae, and J. Kim. 2008. "A new method for generating an invariant iris private key based on the Fuzzy Vault system". *IEEE Transactions on Systems, Man, and Cybernetics- part B: Cybernetics*, vol. 38, n° 5, p. 1302-1313.
- MA, Ferrer, Alonso JB, and Travieso CM. 2005. "Offline Geometric Parameters for Automatic Signature Verification Using Fixed-Point Arithmetic". *IEEE Trans Pattern Analysis Machine Intelligence*, vol. 27, n° 6, p. 993-997.
- Mahamud, S. and M. Hebert. 2009. "The Optimal Distance Measure for Object Detection". In *Int'l Conf. on Computer Vision*. (Kyoto, Japan 2009).
- Maiorana, E, P Campisi, J Fierrez, J Ortega-Garcia, and A. Neri. 2010. "Cancelable templates for sequence-based biometrics with application to on-line signature recognition". *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, vol. 40, n° 3, p. 525-538.
- Meenakshi, V. 2010. "Retina and Iris Based Multimodal Biometric Fuzzy Vault". *Int'l Journal of Computer Applications*, vol. 29, n° 1.
- Meenakshi, V. S. and G. Padmavathi. 2010. "Retina and Iris based multimodal biometric Fuzzy Vault". *Int'l Journal of Computer Applications*, vol. 1, n° 29, p. 67-73.
- Menezes, A, P Van Oorschot, and S Vanstone, 1996. *Hand book of applied cryptography*. CRC press.
- Mohammadi, S. and S. Abedi. 2008. "ECC-Based Biometric Signature: A New Approach in Electronic Banking Security". In *Int'l Symposium on Electronic Commerce and Security*. p. 763-766.
- Nagar, A, K Nandakumar, and A Jain. 2011. "Multibiometric cryptosystems based on feature level fusion". *IEEE Transactions on Information Forensics and Security*, vol. 7, n° 1, p. 255-268.
- Nandakumar, K. 2008. "Multibiometric Systems: Fusion Strategies and Template Security". PhD thesis, Michigan state Univ.
- Nandakumar, K., K. Jain, and S. Pankanti. 2007. "Fingerprint Based Fuzzy Vault: Implementation and Performance". *IEEE Trans. on Information Forensic and Security*, vol. 2, n° 4, p. 744-757.

- Nyang, D. and K. Lee. 2007. "Fuzzy Face Vault: how to implement Fuzzy Vault with weighted features". *Proc. Int'l conf. on universal access in human computer interaction, LNCS*, vol. 4554, p. 491-496.
- Oliveira, L., E. Justino, and R. Sabourin. 2007. "Off-line signature using writer-independent approach". In *Int'l Joint Conference on Neural Networks*. (Orlando, FL, USA 2007), p. 2539-2544.
- Orvos, P. 2002. "Towards biometric digital signatures". In *Net workshop, Eszterhazy College, Eger*. p. 26-28.
- Pekalska, Elzbieta and Robert P.W. Duin. 2002. "Dissimilarity Representations Allow for Building Good Classifiers". *Pattern Recognition Letters*, vol. 23, n° 8, p. 161-166.
- Pekalska, Elzbieta and Robert P.W. Duin. 2005. "The Dissimilarity Representation for Pattern Recognition: Foundations and Applications". *Machine Perception and Artificial Intelligence*, vol. 64.
- Pekalska, Elzbieta and Robert P.W. Duin. 2006. "Dissimilarity-based Classification for Vectorial Representations". In *18th Int'l Conf. on Pattern Recognition*. p. 137-140.
- Pekalska, Elzbieta, Robert P.W. Duin, and Pavel Paclík. 2006. "Prototype Selection for Dissimilarity-based Classifiers". *Pattern Recognition*, vol. 39, n° 8, p. 189-208.
- Ramanan, Deva and Simon Baker. 2011. "Local Distance Functions: A Taxonomy, New Algorithms, and an Evaluation". *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 3, n° 4, p. 794-806.
- Ratha, N. K., J. H. Connell, and R. M. Bolle. 2001. "Enhancing security and privacy in biometrics-based authentication systems". *IBM Systems Journal*, vol. 40, n° 3, p. 614-634.
- Rathgeb, Christian and Andreas Uhl. 2010. "Two-Factor authentication or how to potentially counterfeit experimental results in biometric systems". In *The 7th Int'l conference on Image Analysis and Recognition*. p. 296-305.
- Reddy, E. and I.Babu. 2008. "Performance of Iris based Hard Fuzzy Vault". *Int'l Journal of Computer science and Network security*, vol. 8, n° 1, p. 387-396.
- Reynolds, D.A., Quatieri T.F. Dunn R.B. 2000. "Speaker verification using adapted gaussian mixture models". *Digital Signal Processing*, vol. 10, p. 1-3.
- Rivard, D., E. Granger, and R. Sabourin. 2013. "Multi-Feature extraction and selection in writer-independent offline signature verification". *Int'l Journal on Document Analysis and Recognition*, vol. 16, n° 1, p. 83-103.
- Rivest, R, A Shamir, and L. Adleman. 1978. "A method for obtaining digital signatures and public-key cryptosystems". *Communications of the ACM*, vol. 21, p. 120-12.

- Sabourin, R. and G. Genest. 1994. "An Extended-Shadow-Code based Approach for Off-Line Signature Verification". In *Proc. of the 12th Int'l conf. on Pattern Recognition*. (Jerusalem 1994), p. 450-453.
- Schapire, R. 2002. "The Boosting Approach to Machine Learning: An Overview". In *Proc. MSRI Workshop on Nonlinear Estimation and Classification*.
- Shamir, A. 1979. "How to share a secret". In *Communications of ACM*. (New York, NY, USA 1979), p. 612-613.
- Solar, J.R., C. Devia, P. Loncomilla, and F Concha. 2008. "Off-line Signature Verification using Local Interest Points and Descriptors". In *Lecture Notes in Computer Science*. p. 22-29.
- Soutar, C., D. Roberge, A. Stoianov, R. Golroy, and B. Vijaya Kumar, 1999. *Biometric Encryption, ICSA Guide to Cryptography*. McGraw-Hill.
- Srihari, Sargur, Aihua Xu, and Meenakshi K. Kalera. 2004. "Learning Strategies and Classification Methods for Off-Line Signature Verification". In *Proc. of the 9th Int'l Workshop on Frontiers in Handwriting Recognition*. (Tokyo, Japan 2004), p. 161-166.
- Tieu, K. and P. Viola. 2004. "Boosting image retrieval". *Int'l Journal of Computer Vision*, vol. 56, n° 1, p. 17-36.
- Uludag, U., S. Pankanti, S. Prabhakar, and A.K. Jain. 2004. "Biometric Cryptosystems: Issues and Challenges". *Proc. of the IEEE*, vol. 92, n° 6, p. 948-960.
- Uludag, U., S. Pankanti, and A.Jain. 2005. "Fuzzy vault for fingerprints". In *proc of Audio and video-based biometric person authentication*. (Rye Town, NY 2005), p. 310-319.
- Uludag, Umut. 2006. "Secure biometric systems". PhD thesis, Michigan State University, USA.
- Vargas, J., M. Ferrer, C. Travieso, and J. Alonso. 2007. "Off-line handwritten signature GPDS-960 corpus". In *Int'l Conference on Document Analysis and Recognition*. p. 764-768.
- Vargas, J.F., M.A. Ferrer, C.M. Travieso, and J.B Alonso. 2008. "Offline Signature Verification based on High Pressure Polar Distribution". In *Proc of the 11th Int'l Conference on Frontiers in Handwriting Recognition*. p. 373-378.
- Vielhauer, C., R. Steinrnetz', and A Mayerhofer'. 2002. "Biometric hash based on statistical features of online signatures". In *16th Int'l Conference on Pattern Recognition*. (Quebec, Canada 2002), p. 123-126.
- Wang, Y. and KN. Plataniotis. 2007. "Fuzzy Vault for Face based cryptographic key generation". In *Biometrics Symposium*. (Baltimore, Maryland, USA 2007), p. 1-6.

- Weinberger, K. and L. Saul. 2009. "Distance Metric Learning for Large Margin Nearest Neighbor Classification". *Machine Learning Research*, vol. 10, p. 207-244.
- Yager, N. and T. Dunstone. 2010. "The Biometric Menagerie". *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, n° 2, p. 220-230.
- Yang, S. and I. Verbauwhede. 2005. "Automatic Secure fingerprint verification system based on fuzzy vault scheme.". *IEEE Int'l Conference on Acoustics, Speech, and Signal Processing*, vol. 5, p. 609-612.
- Yip, W. K., A. Goh, D. C. L. Ngo, and A. B. J. Teoh. 2006. "Generating a replaceable cryptographic keys from dynamic handwritten signatures". *Lecture Notes in Computer Science*, vol. 3832, p. 509-515.